Workshop

Future Directions in Systems and Control Theory

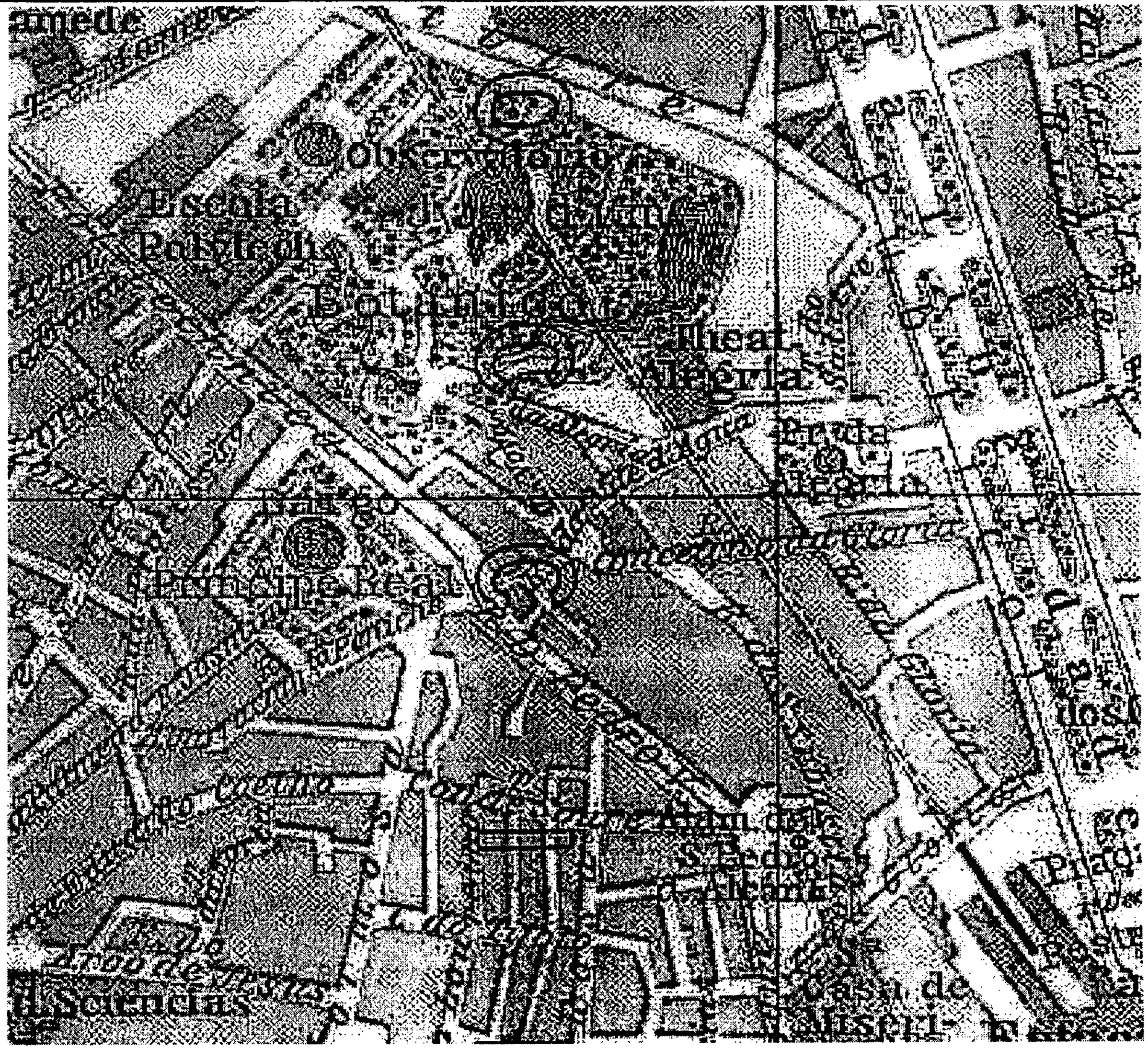
Wednesday , June 23

Viewgraphs - Volume 2

Estimation of Large, Dynamic Processes (With Application to Ground Traffic Control)

Estimation Over Dynamic Graphs

Dr. Robert R. Tenney
ALPHATECH, Inc.
50 Mall Road
Burlington, MA 01803
(781) 273-3388
tenney@alphatech.com



Estimation of Large, Dynamic Processes (With Application to Ground Traffic Control)

Problem: Maintain continuous track of ground vehicles

Solution: Estimation over dynamic graphs

- Tracking vehicles: estimation over time with sequential structures
- Estimating roads: estimation over space with cyclic structures
- Tracking maps: estimation over time and space with approximate algorithms

Conclusion:

- Algorithms that perform estimation over graphs do scale to large problems

Hypothesis:

- Efficiency can be enhanced with discrete event control algorithms
  - Estimation "kernels" form the plant
  - Control algorithms determine sequencing of operations

Problem: Ground Traffic Control

Maintain continuous track of ground vehicles:

- 2500 km^2 area
- >1000 vehicles
- roads poorly mapped

States:

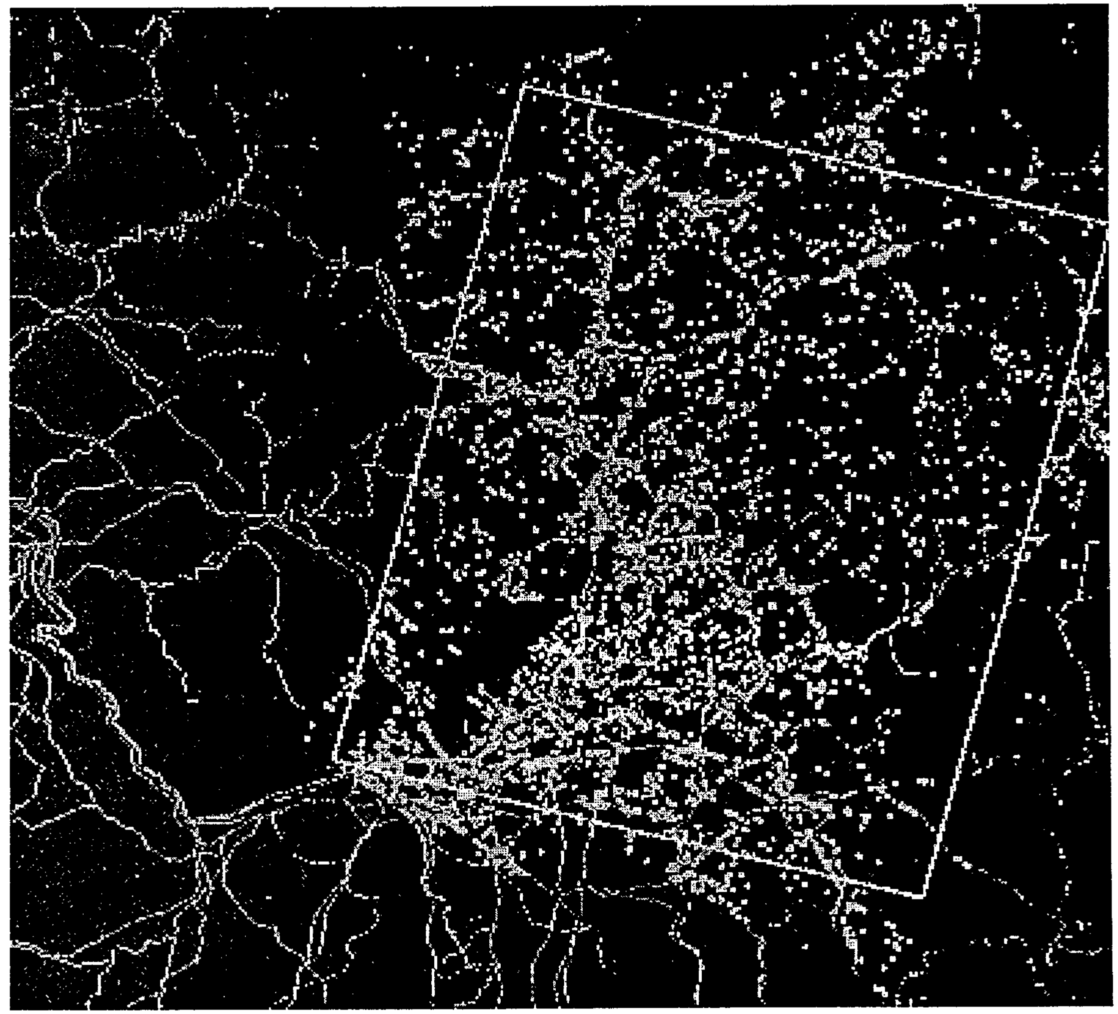
- vehicle positions, velocities
  - current estimate of situation
- road network geometry
  - major impact on vehicle dynamics

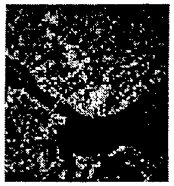
Observables:

- radar detections
- video images

Products:

- estimates of traffic flow
  - densities
  - transit times
- locations of key vehicles





Problem: Ground Traffic Control Has Inhomogeneous Background

Air traffic control

Dynamics:
- Piecewise constant velocity
- Occasional banks and turns

Physical constraints:
- Altitude > ground surface elevation
- Weather avoidance
- Airspace policy

Observables:
- Regular radar/transponder reports
- Few false alarms

Solution technology:
- Hybrid state filtering
  - Detection of maneuvers
  - Estimation of trajectory given maneuver estimates

Ground traffic control

Dynamics
- Piecewise constant acceleration
- Frequent turns, starts, and stops

Physical constraints:
- Altitude = ground surface elevation
- Vehicle avoidance
- Traffic regulations

Observables:
- Irregular radar/video detections
- Many false alarms

Solution technology I (map well known):
- Hybrid state filtering
  - Detection of maneuvers
  - Estimation of trajectory given maneuver estimates

Solution technology II (map poorly known):
- ?

Dynamic Estimation: Tracking Is Part of Ground Traffic Control

Given:

- A discrete time (asynchronously sampled) Markov process:
  - State $x(t_n)$
    - 2D position
    - 2D velocity (speed, heading)
  - State transition probability distribution $p(x(t_n) \mid x(t_{n+1}))$
  - Markov conditional independence property
  - x discrete, continuous, or both
- A series of measurements at sample times:
  - Measurement $y(t_n)$
    - 2D position
    - radial velocity
  - Observation probability distribution $p(y(t_n) \mid x(t_n))$
  - Markov conditional independence property
  - y either discrete, continuous, or both

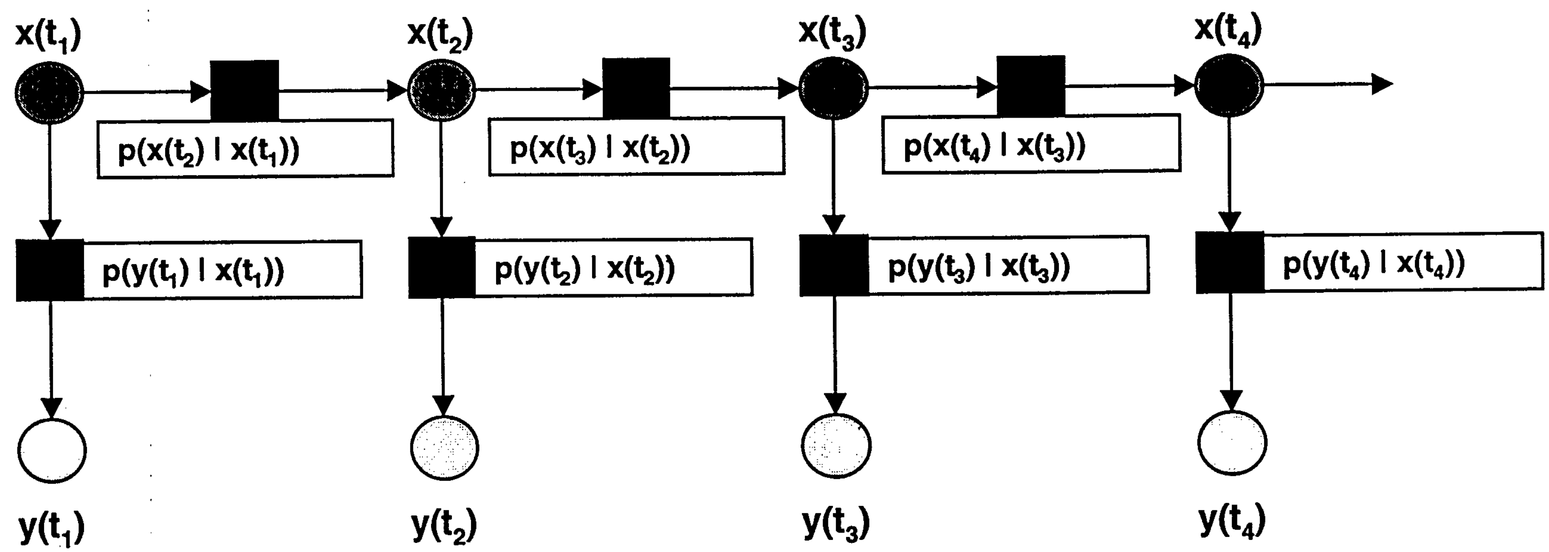
Find:

- Tracking problem:
  - $p(x(t_n) \mid y(t_1), y(t_2), \ldots, y(t_n))$
- Smoothing problem:
  - $p(x(t_1), x(t_2), \ldots, x(t_n) \mid y(t_1), y(t_2), \ldots, y(t_n))$

Dynamic Estimation:
Tracking Uses a Sequence of Random Variables

Causal relations among variables can be displayed as a graph:

- Nodes: represent elemental random variables (states and measurements)
- Arcs: represent conditional probability distributions among variables
- Non-arcs: represent conditional independence assumptions



Also known as Bayes' nets (although this talk makes explicit the conditional pdf's on arcs)

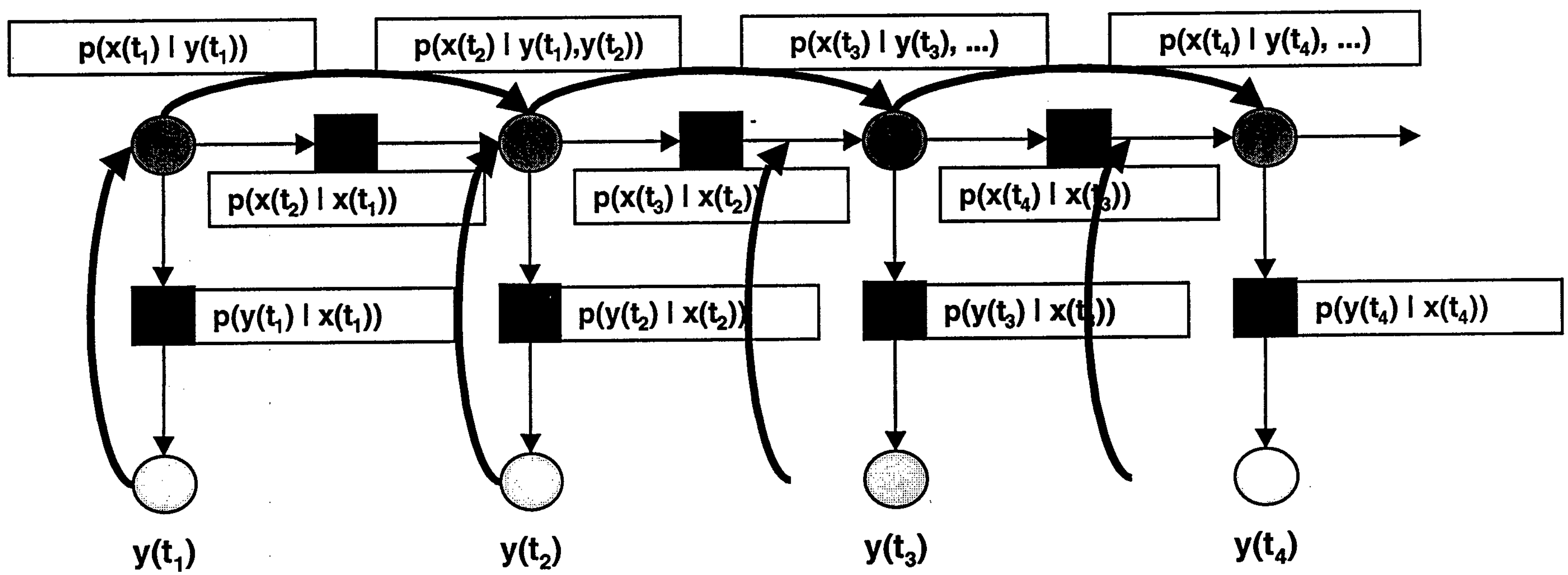
Dynamic Estimation: Tracking Algorithms Propagate Information

Algorithms mirror graph structure:

- **Nodes:** store conditional distributions on elemental random variables
- **Arcs:** transmit likelihood/prediction information among nodes

Elemental operations:

- **Update:** $p(x(t_n) \mid y(t_n), \ldots) = p(y(t_n) \mid x(t_n))\, p(x(t_n) \mid \ldots) / p(y(t_n))$
- **Predict:** $p(x(t_n) \mid y(t_{n-1}), \ldots) = \Sigma\{p(x(t_n) \mid x(t_{n-1}))\, p(x(t_{n-1}) \mid y(t_{n-1}), \ldots)\}$



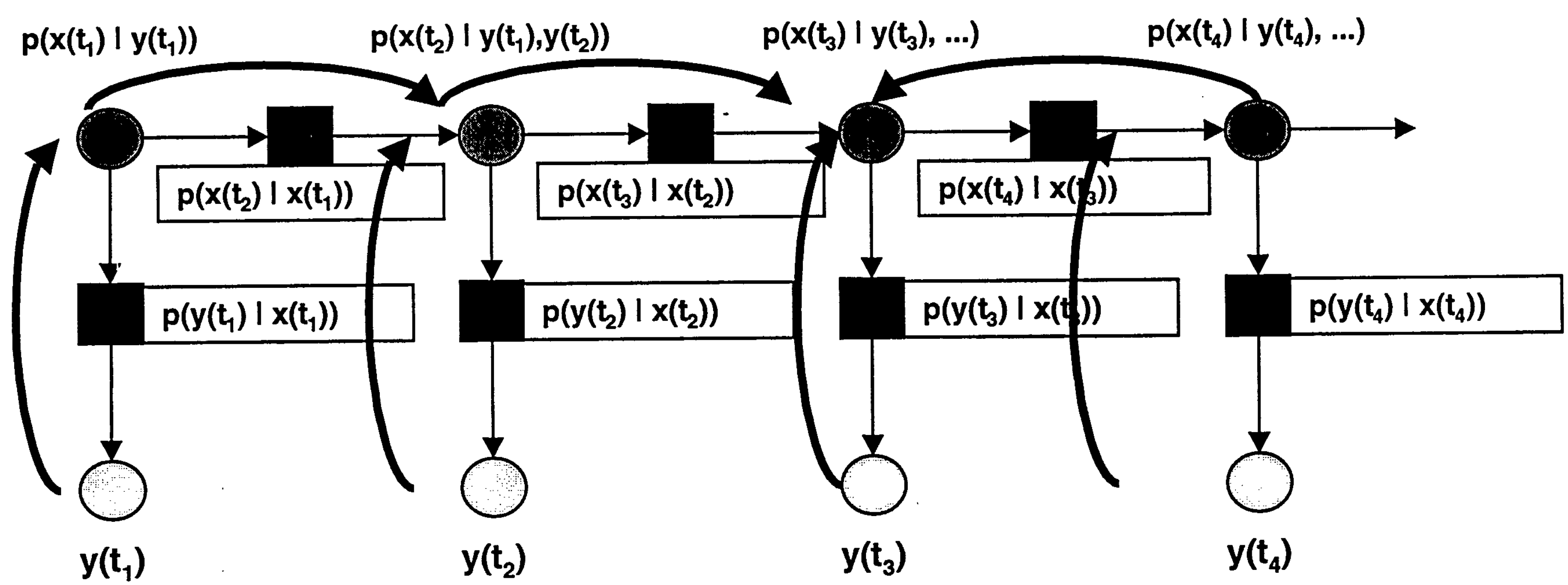
Dynamic Estimation: Smoothing Algorithms Propagate More Information

Algorithms mirror graph structure:

- **Nodes:** store conditional distributions on elemental random variables
- **Arcs:** transmit likelihood/prediction information among nodes

Elemental operations:

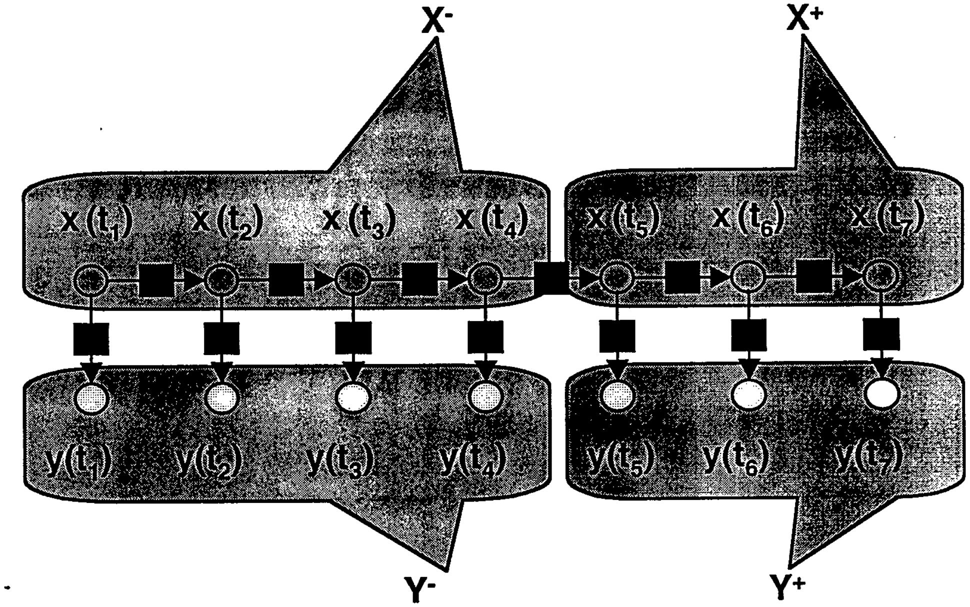
- **Back-update:** $p(x(t_3) \mid y(t_4)) = p(x(t_3)) \Sigma\{p(x(t_4) \mid x(t_3))\, p(x(t_4) \mid y(t_4))/p(y(t_4))\}$
- **Combine:** $p(x(t_3) \mid y(t_1), \ldots, y(t_4)) = p(x(t_3) \mid y(t_1), \ldots, y(t_3))\, \Sigma\{\bullet\}$



Dynamic Estimation: Decomposition Can Be Generalized

Graphical Separation Theorem:

- Assuming conditional independence among variables not connected by arcs, estimation of x(n) separates into:
  - Estimation of $x(t_n)$ given $y(t_1)\ldots y(t_n)$
  - Estimation of $x(t_n)$ given $y(t_{n+1})\ldots y(t_N)$
  - Fusion of the results

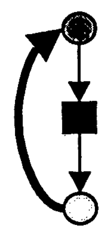


Math:

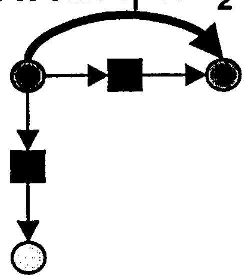
$$
\begin{aligned}
p\left(x(t_n) \mid Y^+, Y^-\right) &= \sum_{X^+, X^-} p\left(x(t_n), X^+, X^- \mid Y^+, Y^-\right) \\
&\propto \sum_{X^+, X^-} p\left(Y^+, Y^- \mid X^+, X^-, x(t_n)\right) p\left(X^+, X^-, x(t_n)\right) \\
&= \sum_{X^+, X^-} \left[p\left(Y^+ \mid X^+, x(t_n)\right) p\left(Y^- \mid X^-, x(t_n)\right)\right]\left[p\left(X^+ \mid x(t_n)\right) p\left(X^- \mid x(t_n)\right) p\left(x(t_n)\right)\right] \\
&= p\left(x(t_n)\right)\left[\sum_{X^+} p\left(Y^+ \mid X^+, x(t_n)\right) p\left(X^+ \mid x(t_n)\right)\right]\left[\sum_{X^-} p\left(Y^- \mid X^-, x(t_n)\right) p\left(X^- \mid x(t_n)\right)\right] \\
&= p\left(x(t_n)\right) p\left(Y^+ \mid x(t_n)\right) p\left(Y^- \mid x(t_n)\right) \\
&\propto p\left(Y^+ \mid x(t_n)\right) p\left(x(t_n) \mid Y^-\right)
\end{aligned}
$$

Dynamic Estimation: Algorithm Handles Out-Of-Sequence Measurements

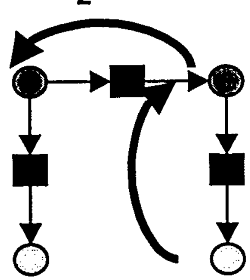
Update at t_1:



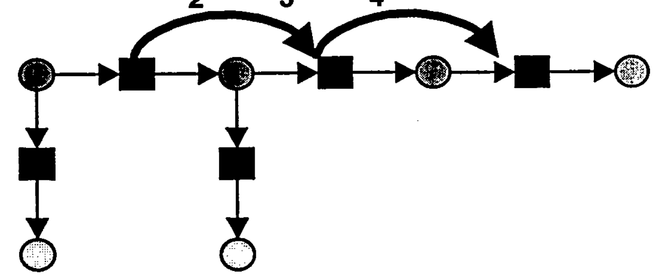
Predict from t_1 to t_2:



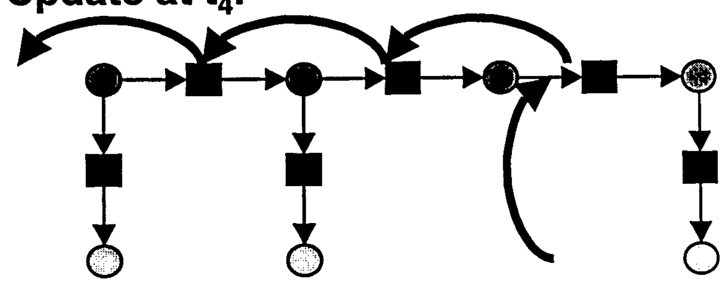
Update at t_2:



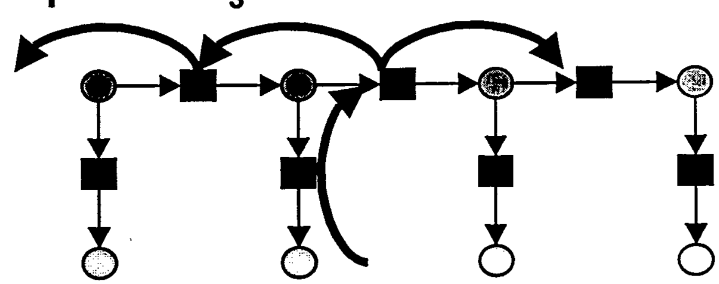
Predict from t_2 to t_3 to t_4:



Update at t_4:



Update at t_3:



Dynamic Estimation: Graph Models Provide a Scalable Scalable Approach

Each node represents a random variable:

- Stores various conditional probability distributions
- Initialized by prediction operation
  - –Construct the node
  - –Use Chapman-Kolmogorov equation to predict current distribution
- Updated by measurements
  - –If directly observed, set to observed value
  - –If indirectly observed, use Bayes' rule (perhaps with adjoint C-K operation) to update

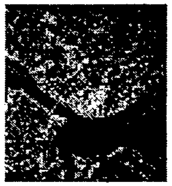
Models reside on arcs of the graph:

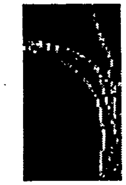
- Each arc represents a conditional pdf
  - –Between random variables at its endpoints
  - –Directionality distinguishes conditioning and conditioned variables
- Absent arcs represent conditional independence assumptions

Algorithms move information between nodes:

- –C-K if information flow aligns with direction of link
- –Bayes or C-K* if information flow opposes direction of link

Dynamic Estimation: Graph Algorithms Provide a Scalable Approach



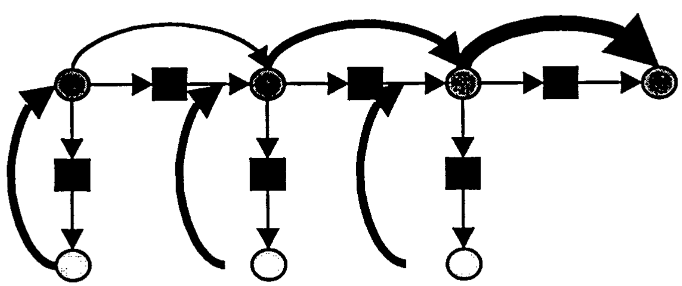


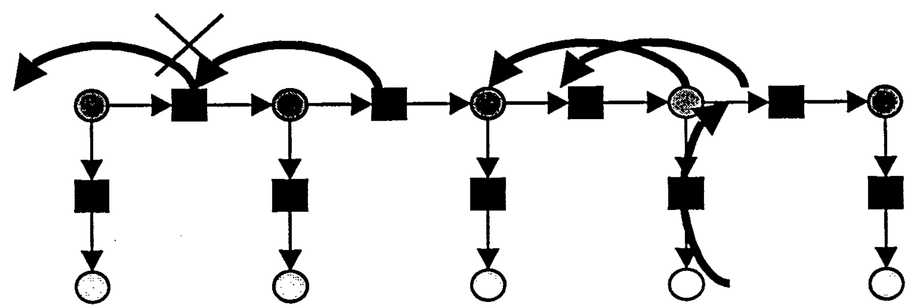
Scalability:

- No consolidated storage of pdfs
  - $p(x_1, x_2, x_3 \mid y)$ stored in factored form
    - conditional distributions on arcs (fixed)
      - $p(x_3 \mid x_2)$, $p(x_2 \mid x_1)$
    - conditional estimates at nodes
      - $p(x_1 \mid y)$, $p(x_2 \mid y)$, $p(x_3 \mid y)$

- Asynchronous, incremental manipulation of pdfs
  - At nodes
  - Across arcs

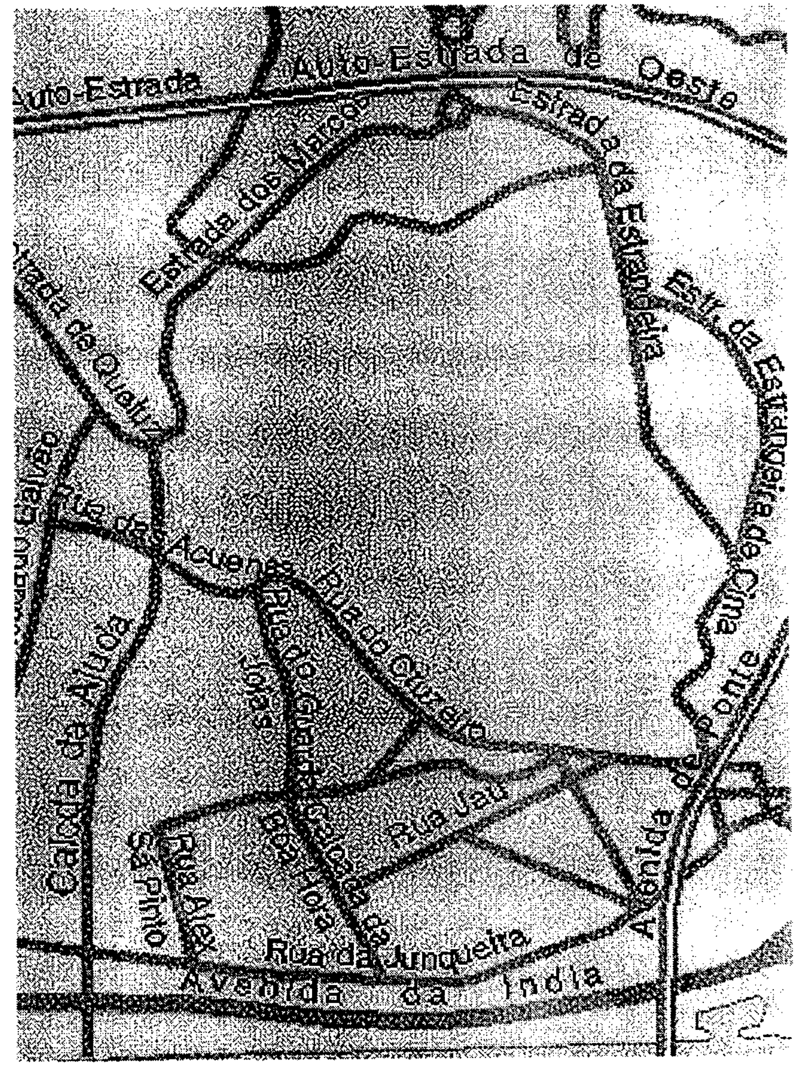
- Potential for reduction of incremental operations
  - Batching
  - Incomplete updates





Spatial Estimation:
Roads Strongly Influence Ground Vehicle Motion

$\neq$



Spatial Estimation: Road Shape Estimation Is Part of Ground Traffic Control

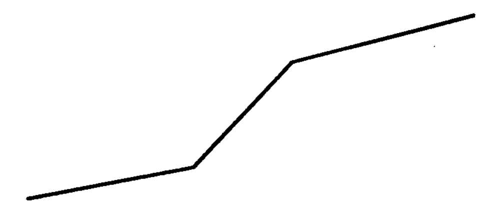
Given:

- A discrete (adaptively sampled) spatial Markov process:
  - State $x(s_n)$
    - 2D position of centerpoint
    - length, orientation
  - Neighboring state probability distribution
  - Markov conditional independence property
  - x discrete, continuous, or both
- A series of measurements of states:
  - Measurement $y(s_n)$
    - 2D position
    - orientation
  - Observation probability distribution $p(y(s_n) \mid x(s_n))$
  - Markov conditional independence property
  - y either discrete, continuous, or both

Find:

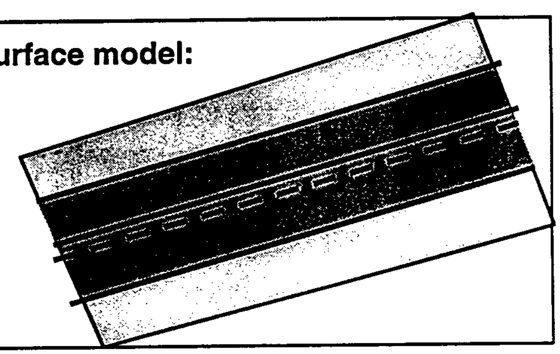
- Smoothing problem:
  - $p(x(s_1), x(s_2), \ldots, x(s_n) \mid y(s_1), y(s_2), \ldots, y(s_n))$

Centerline model:



Boundary model:

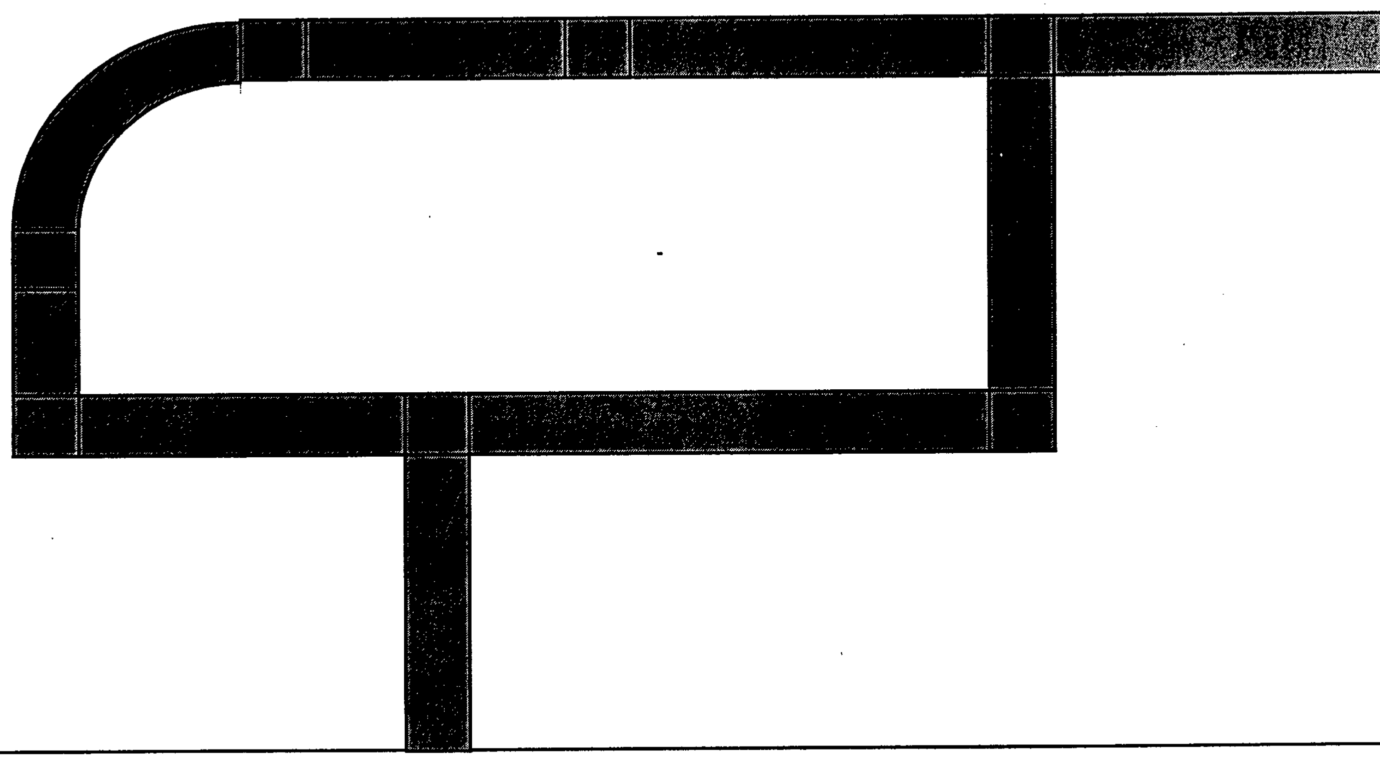
Surface model:



Spatial Estimation: Roads Form Networks

Road networks can be displayed as a graph:

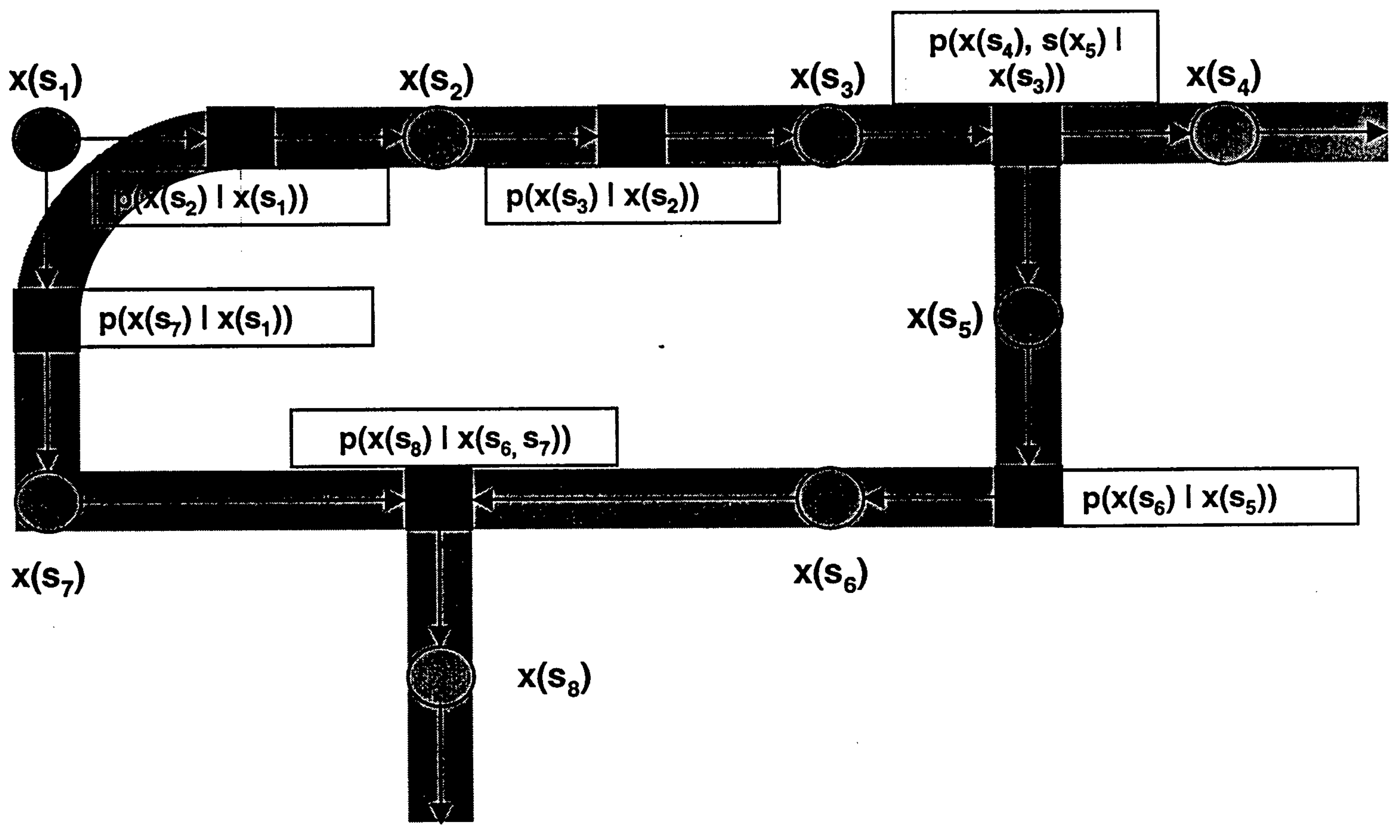
- Segments: linear (arc, spline) approximations to road shape
- Intersections: connections among segments



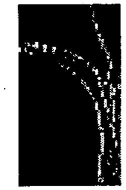
Spatial Estimation:
Graphs Models Form Networks

Road network estimates can be displayed as a graph:

- **Nodes:** represent estimates of segment location, length, orientation
- **Arcs:** represent conditional dependencies imposed by intersections
- **Non-arcs:** represent conditional independence assumption

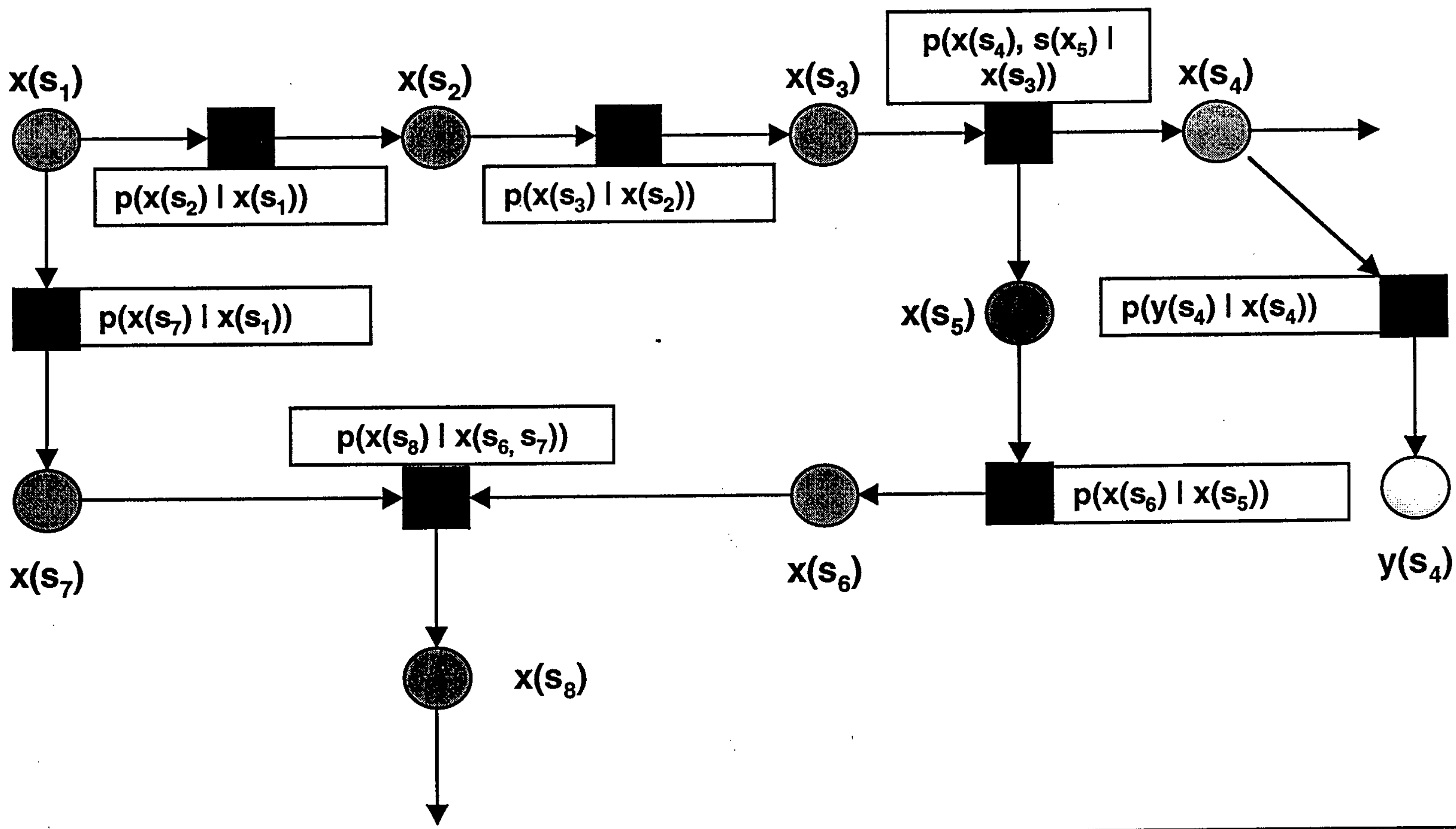


Spatial Estimation: Graph Models Can Be Updated



Measurements provide information about individual segments:

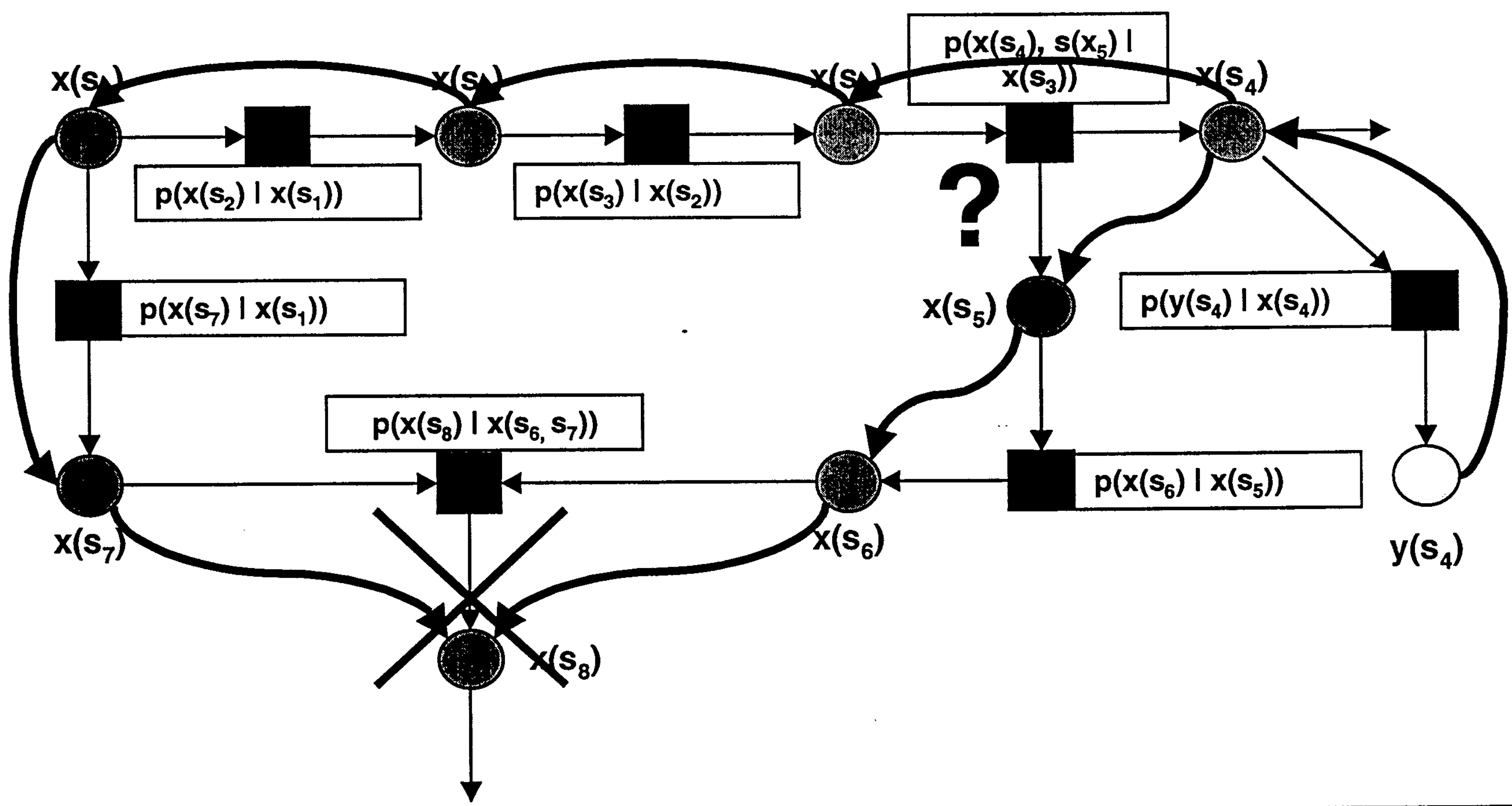
- Location, length, orientation
- Corrupted by noise



Spatial Estimation: General Graph Updates Become Complex

Complications arise due to cycles:

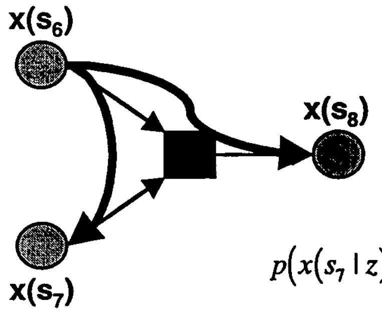
- More complex topology between nodes
- Cycles violate Graph Separation Theorem



Spatial Estimation: Extend Operators to Include Multi-node Relations

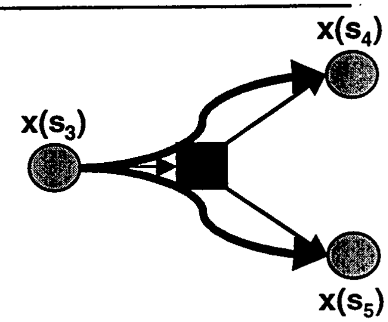
For three-way relations, local merge suffices:

- **Use conditional distribution on source to update/predict joint pdf on others**
- **Marginalize the result**



$$p(x(s_7 \mid z)) = p(s_7) \int_{s_6, s_8} p(s_8 \mid s_6, s_7) p(s_6 \mid z)$$

$$p(x(s_8 \mid z)) = \int_{s_6, s_7} p(s_8 \mid s_6, s_7) p(s_6 \mid z) p(s_7)$$

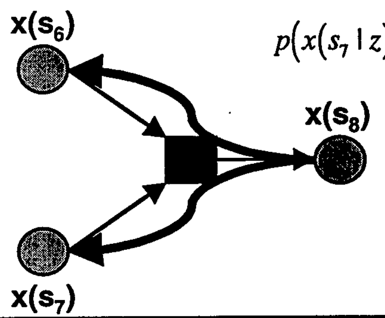


$$p(x(s_4 \mid z)) = \int_{s_3, s_5} p(s_4, s_5 \mid s_3) p(s_3 \mid z)$$

$$p(x(s_5 \mid z)) = \int_{s_3, s_4} p(s_4, s_5 \mid s_3) p(s_3 \mid z)$$

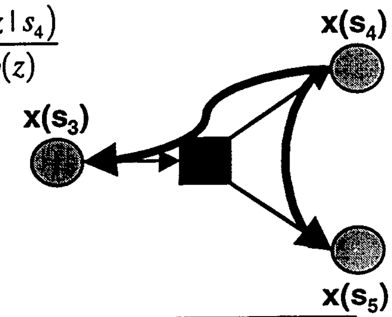
$$p(x(s_6 \mid z)) = p(s_6) \int_{s_7, s_8} p(s_8 \mid s_6, s_7) p(s_7) \frac{p(z \mid s_8)}{p(z)}$$

$$p(x(s_7 \mid z)) = p(s_7) \int_{s_6, s_8} p(s_8 \mid s_6, s_7) p(s_6) \frac{p(z \mid s_8)}{p(z)}$$



$$p(x(s_3 \mid z)) = p(s_3) \int_{s_4, s_5} p(s_4, s_5 \mid s_3) \frac{p(z \mid s_4)}{p(z)}$$

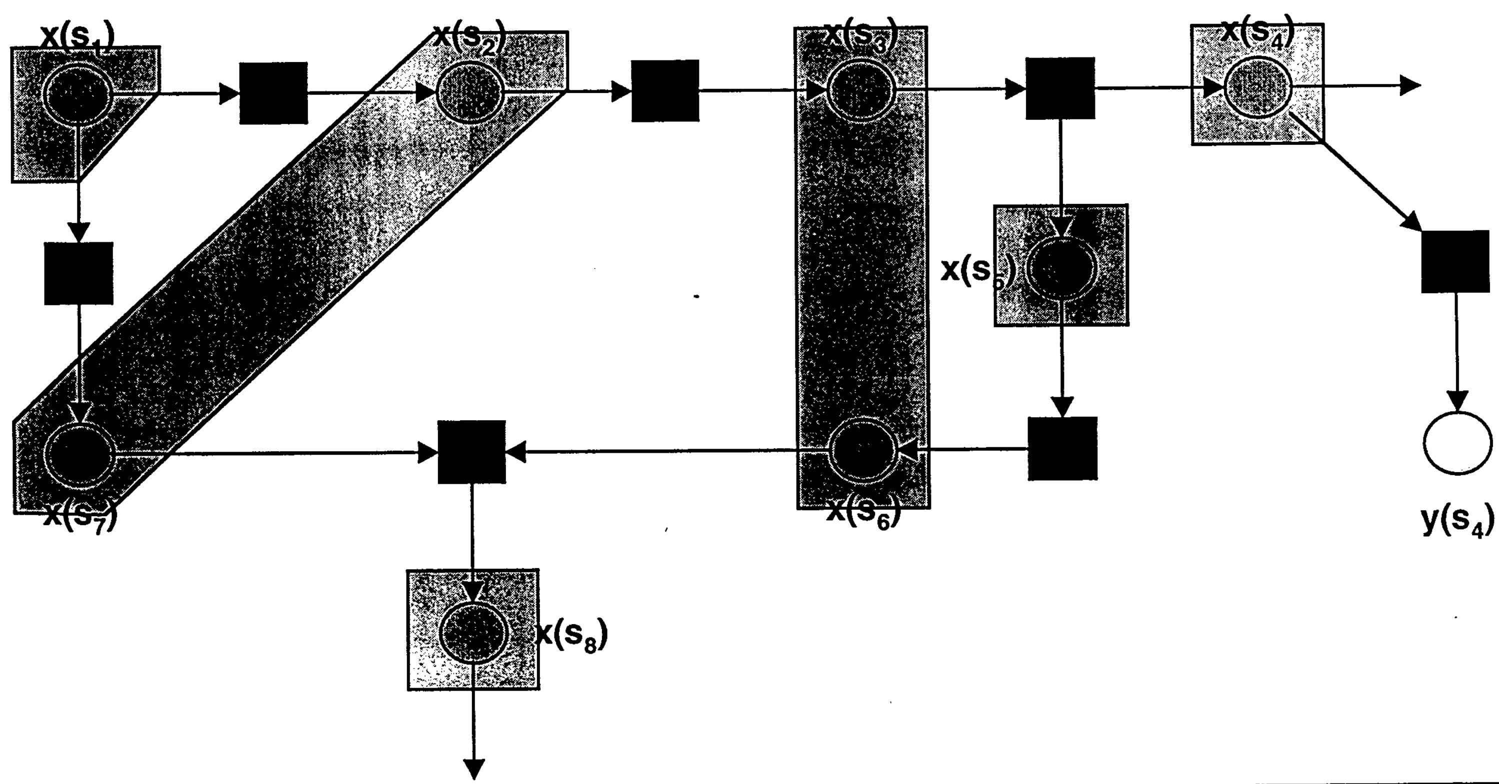
$$p(x(s_5 \mid z)) = \int_{s_3, s_4} p(s_4, s_5 \mid s_3) p(s_3) \frac{p(z \mid s_4)}{p(z)}$$



Spatial Estimation: Topological Algorithms Restore Acyclic Structure

Combining nodes can eliminate (undirected) cycles:

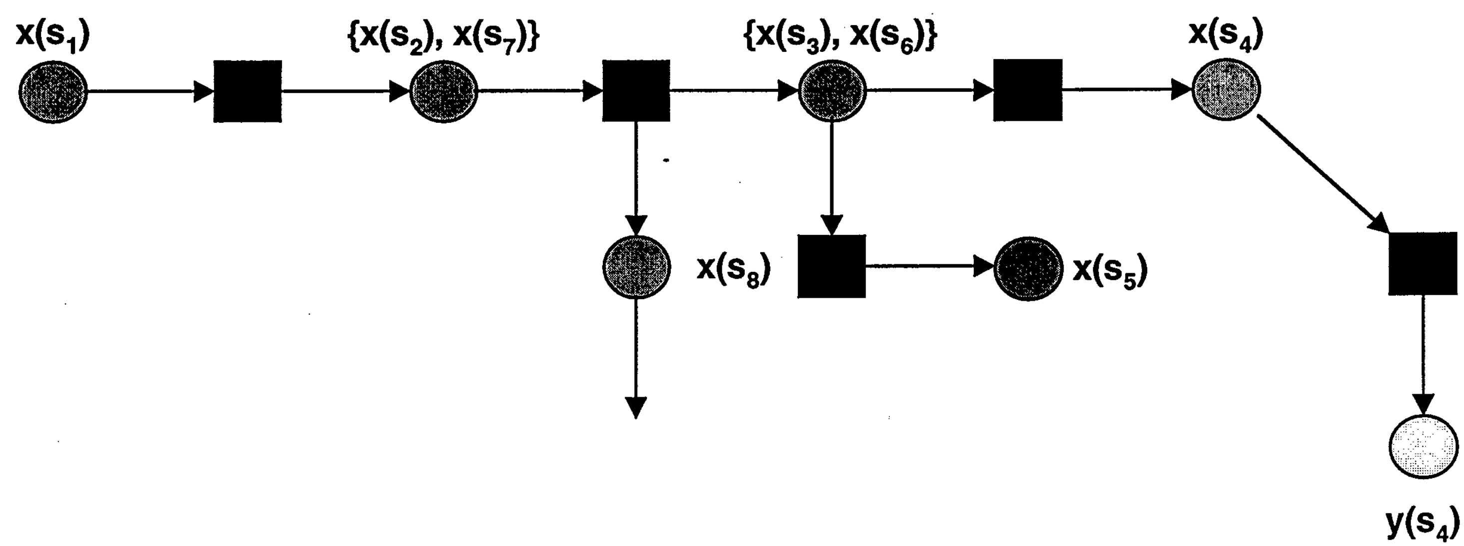
- Graph Separation Theorem holds for any cut set of a graph
- Merging nodes on cutsets eliminates cycles
- Exact algorithms work on reduced graph



Spatial Estimation: Exact Algorithms Operate on Acyclic Structure

Merging nodes on cut sets maintains conditional independence:

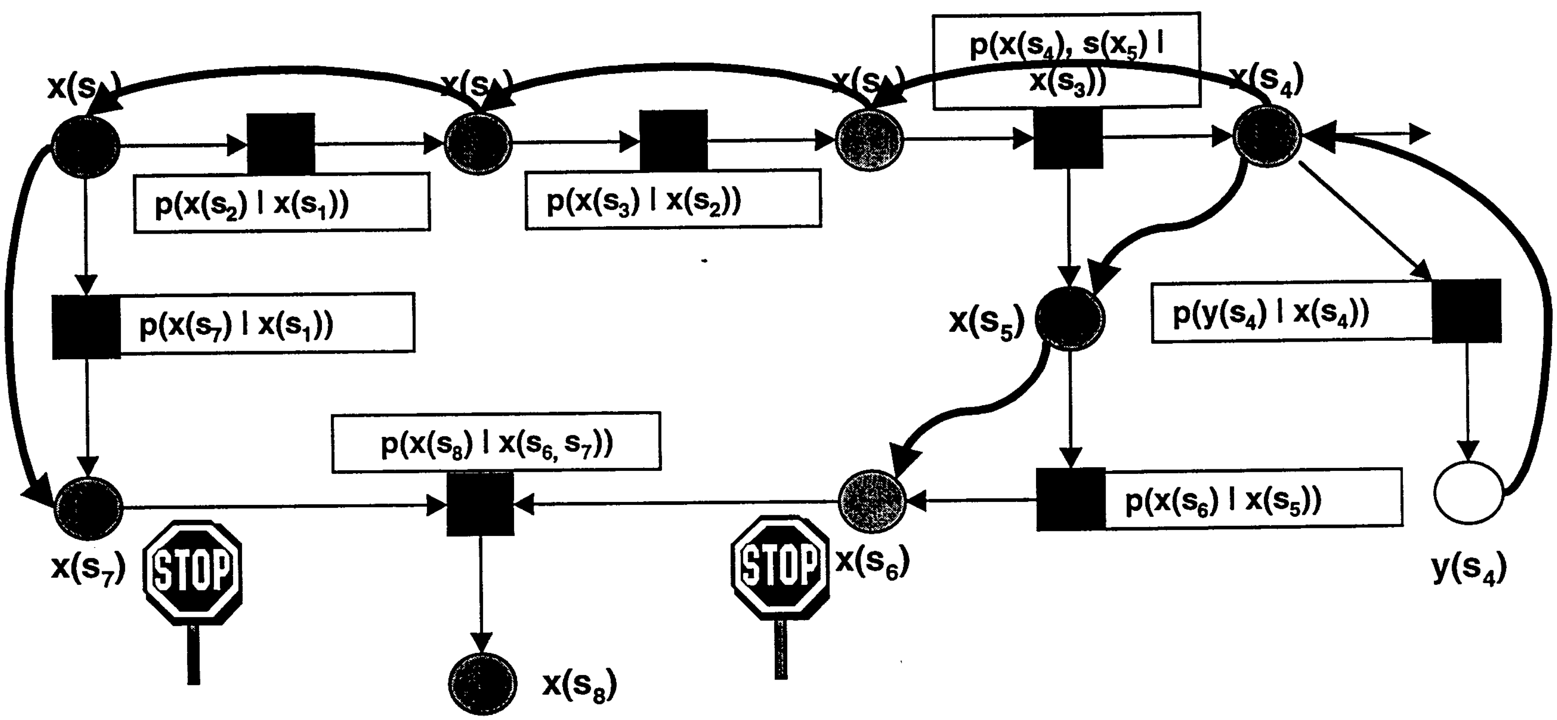
- Complexity of individual nodes increases
- Complexity of conditional distributions on arcs increases
- Does not scale well for road networks, especially in urban areas



Spatial Estimation: Approximate Algorithms Limit Propagation

Two costs of approximation:

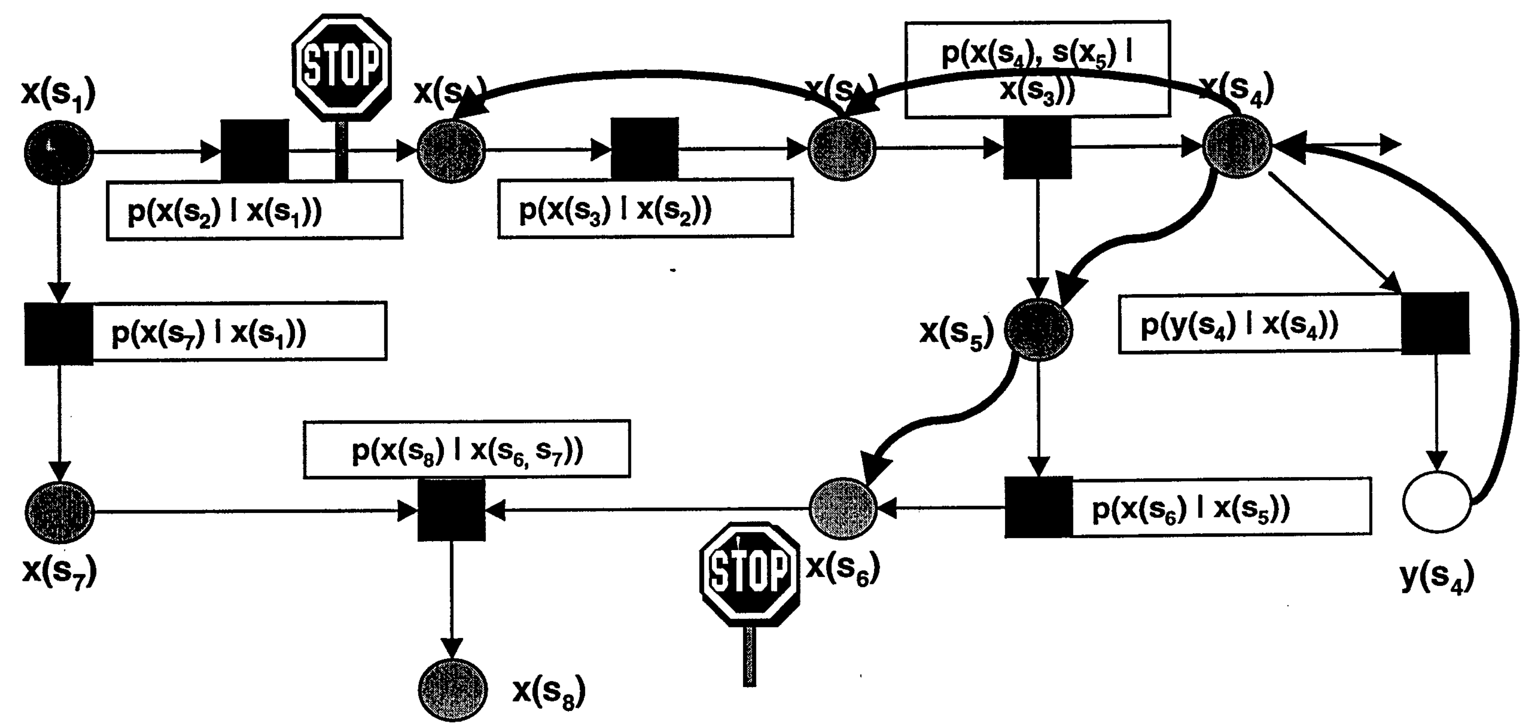
- **Suboptimal use of information - some estimates fail to get updated**
  - –Slower convergence for given data; more data required to achieve given accuracy
- **Use of incorrect model adds errors - conditional independence implicit**
  - –Adjustments (diffusion) to resulting PDF required; slower convergence



Spatial Estimation: Approximate Algorithms Enhance Scalability

Law of diminishing returns:

- Successive adjustments (updates, predicts) have diminishing impact on result
  - –Smaller reduction in entropy (or variance)
  - –Rate of decrease depends on mutual information of adjacent nodes
- Scalable algorithms terminate propagation for each update



Spatial Estimation: Graph Algorithms Provide a Scalable Approach

Each node represents a random variable:

- Stores various conditional probability distributions
- Initialized by prediction operation
  - –Construct the node
  - –Use Chapman-Kolmogorov equation to predict current distribution
- Updated by measurements
  - –Richer set of primitive update/predict operations
  - –All operate on the neighborhood of a node

Models reside on arcs of the graph:

- Each arc represents a conditional pdf
  - –Between random variables at its endpoints
  - –Directionality distinguishes conditioning and conditioned variables
- Absent arcs represent conditional independence assumptions

Algorithms move information between nodes:

- –Bayes, C-K, C-K*, or combinations thereof
- –Termination required for efficiency

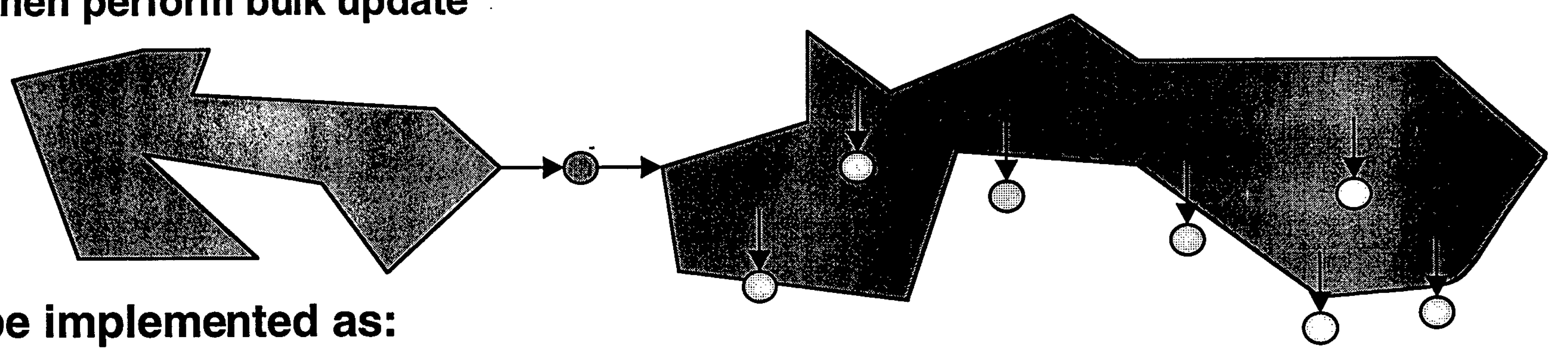
Spatial Estimation: Algorithm Control Can Be Helpful

Candidate criteria for terminating updates:

- Limit to N hops, for some N
- Limit to boundaries of acyclic subgraph rooted at source of observation
- Limit to updates that result in less than ε reduction in entropy

Batch processing is possible:

- Accumulate updates from one direction until sufficient information gain has been achieve, then perform bulk update



Both can be implemented as:

- Elements of local, asynchronous computations at a node
  - Proof of global stability required
- A centralized supervisor that allocates computation to most useful operations
  - Proof of scalability required

Dynamic Spatial Estimation: Ground Traffic Control Operates Over Time and Space

Problem:

- Maintain track of groups of vehicles traveling over poorly known terrain

State variables:

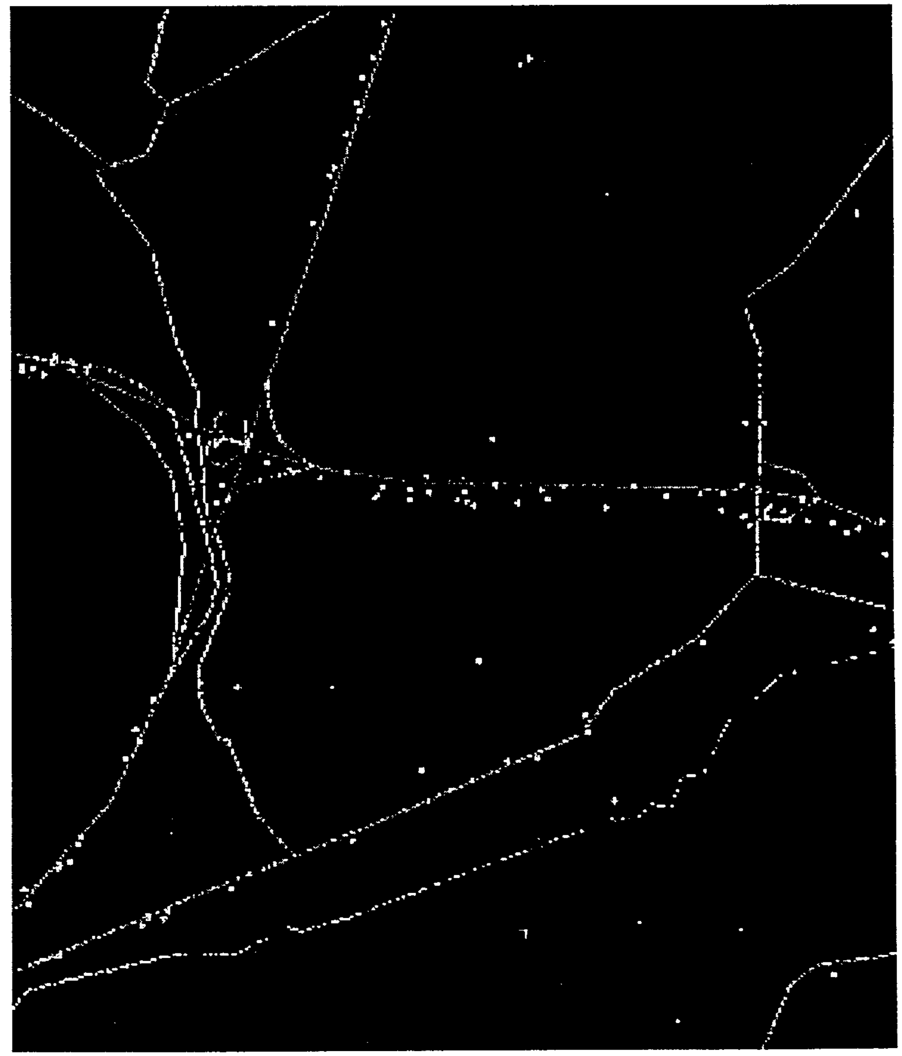
- Positions of vehicles over time
- Shape of road network

Observations:

- Detections of vehicles over time
  - –Radar: wide area, poor resolution
  - –Video: narrow area, good resolution

Approach:

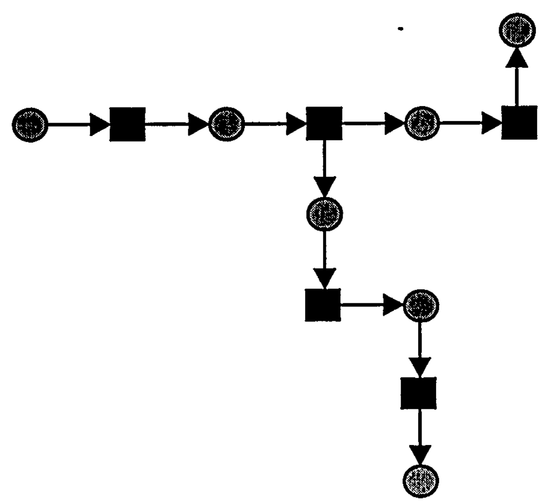
- Dynamic graph algorithms



Dynamic Spatial Estimation: Initial Conditions Consist of a (Poor) Map

Map consists of a segmentation of the road network

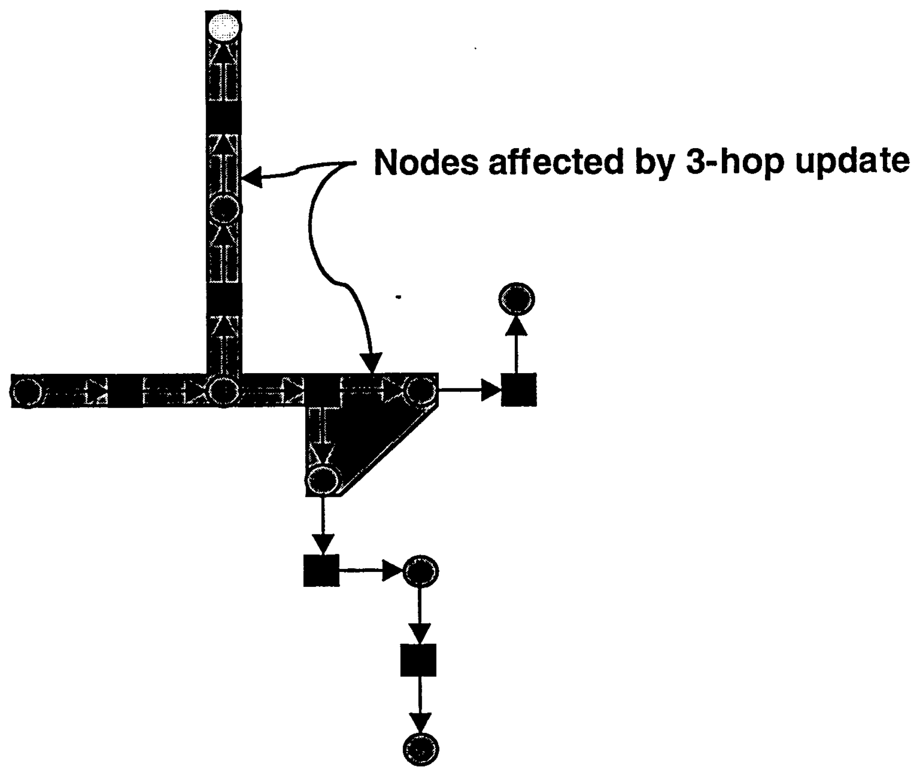
- Segmentation is fixed
  - –Variable segmentation requires additional control
- Parameters of each segment contain errors
  - –Correlation only among neighbors



Dynamic Spatial Estimation: First Vehicle Appears, Gets Observed

State consists of two parts:

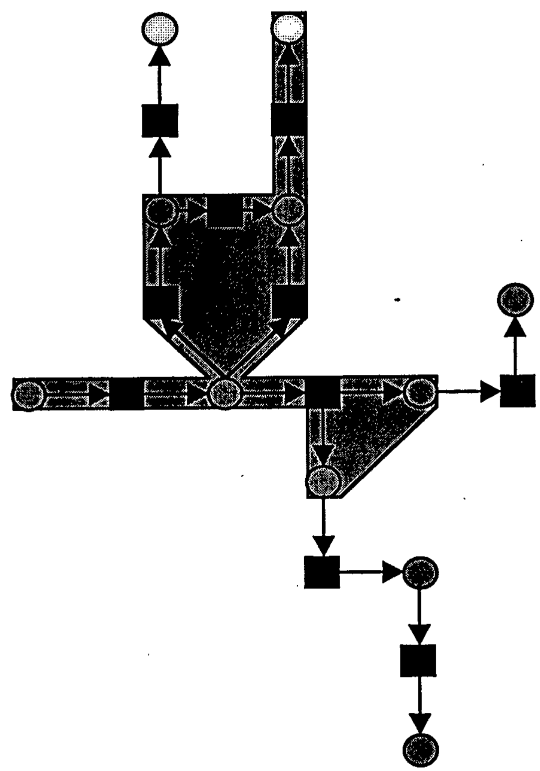
- Segment occupied by vehicle, represented by arc in graph
  - –Construction of assignment requires additional control
- Location/speed of vehicle on that segment
  - –one-dimensional position/speed (if vehicle stays on road)



Dynamic Spatial Estimation: First Vehicle Moves, Gets Observed

Assume vehicle stays on same segment:

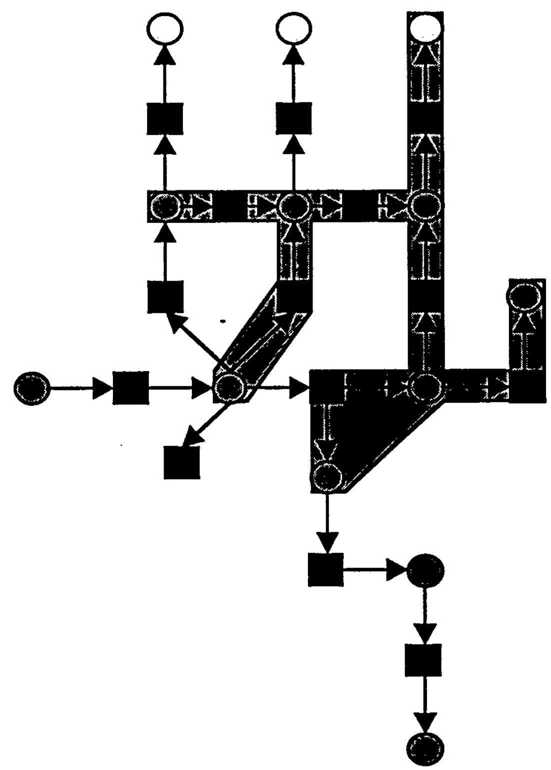
- Decision to stay on segment requires additional control
- Introduces cycle between vehicle positions and road segment parameters



Dynamic Spatial Estimation: First Vehicle Moves Again, Gets Observed

Assume vehicle moves to new segment:

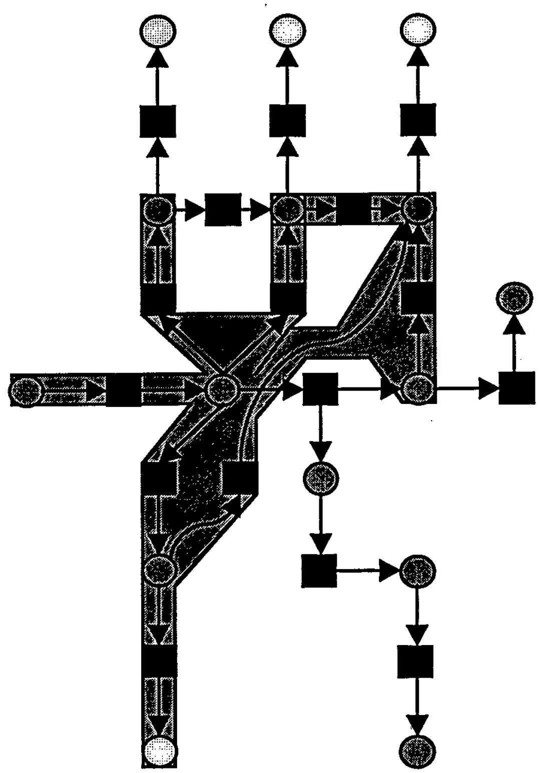
- Decision to change segments requires additional control
- Introduces cycle between vehicle positions and road segments



Dynamic Spatial Estimation: Second Vehicle Appears, Gets Observed

Assume this is a different vehicle:

- **Decision to start a new track requires additional control**
- **Update includes expected inter-vehicle spacing statistics**



Dynamic Spatial Estimation: The Update Process Dynamically Maintains Consistency

Two levels of dynamics:

- Topological:
  - –Addition of new nodes and arcs, cued by new detection reports
  - –(Deletion of outdated estimates, cued by update rate and priority)
    - Decision to delete requires additional control
- Statistical:
  - –Likelihood functions and updated distributions flow among nodes
  - –Local computations at nodes perform updates, prepare new likelihoods and posteriors

Conditional distributions on arcs induce the update algorithm to maintain consistency among estimates:

- Road segments connect to one another
- Vehicles stay on roads
- Vehicle spacing is realistic
- Vehicle positions match observations

Dynamic Spatial Estimation: Dynamics Differ Among Classes Of Nodes

Measurement nodes:
- Always known
- Never updated
- No convergence

Vehicle position nodes:
- Never known perfectly
- Updated over window of a few minutes
- Process noise limits convergence

Road network nodes:
- Never known perfectly
- Updated forever
- Absence of process noise permits convergence to zero variance (theoretically)

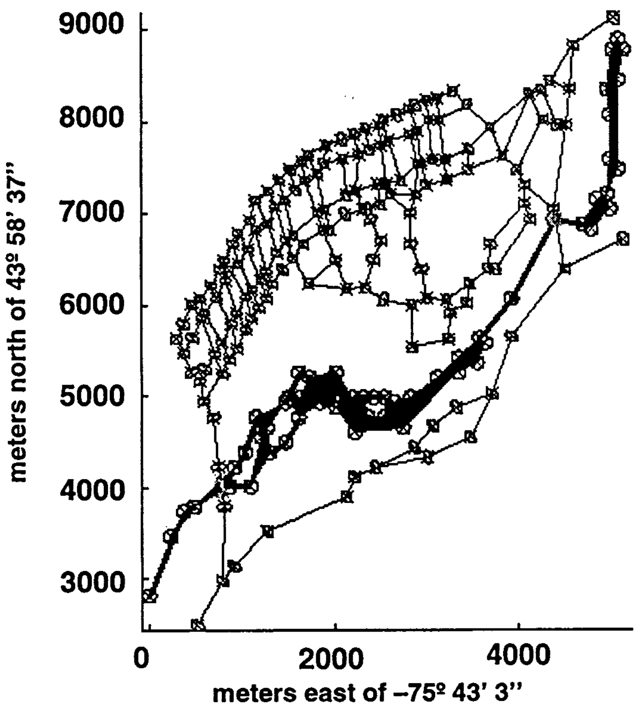
Dynamic Spatial Estimation: Time–scale Decomposition Is Feasible

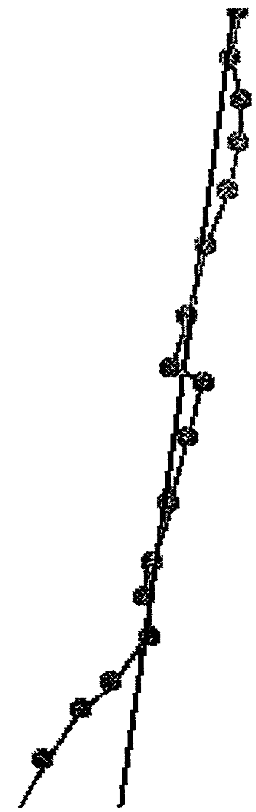
Slow time scale:

- Assume all vehicles are on roads
- Assume detections are of points on roads, not vehicles
- Estimate road network from these detections

Fast time scale:

- Assume roads are fixed
- Assume detections are of vehicles, not roads
- Estimate vehicle trajectories, subject to road constraints





Conclusion:
Dynamic Graph Estimation Solves Large Scale Problems

Topology reflects conditional independence, defines solution structure

- Decomposition of posterior pdf into related factors
- Statistical relations between neighboring nodes

Nodes maintain actual estimates

- Update from posterior/likelihood functions from neighbors
- Generate posterior/likelihood functions to neighbors

Update computations are naturally distributed and asynchronous

- Propagate posterior/likelihood messages between nodes
- Terminate propagation when value-added is insignificant

Topological dynamics are event-driven:

- Receipt of a new report
- Out-of-memory exception

Hypothesis: Dynamic Graph Algorithms Need Control

Adaptive update propagation

- Choose between cycle merging and splitting
- Determine radius of propagation
- Consolidate many small updates into a single update

Topology control:

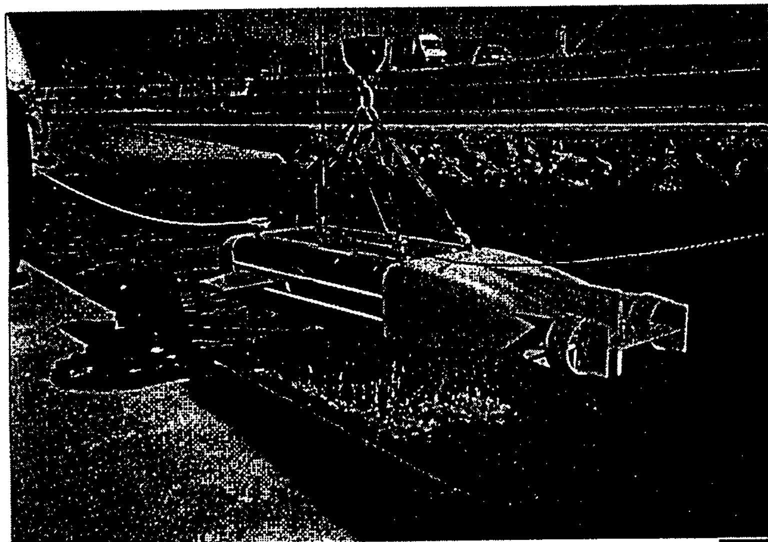
- Multiple assignment hypotheses:
  - Detections to vehicles
  - Vehicles to road segments
- Adaptive sampling
  - Of temporal trajectories
  - Of spatial boundaries
- Multiple resolutions
  - Temporary interpolation of coarse-to-fine scales
  - Persistent aggregation of fine-to-coarse information

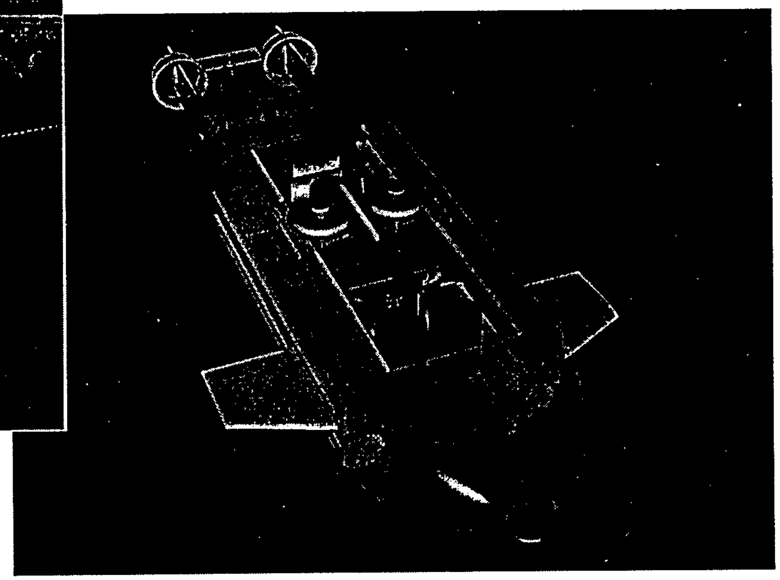
Linear and Nonlinear Control Theory with applications to Navigation, Guidance and Control of Robotic Air and Ocean Vehicles

Antonio M. Pascoal

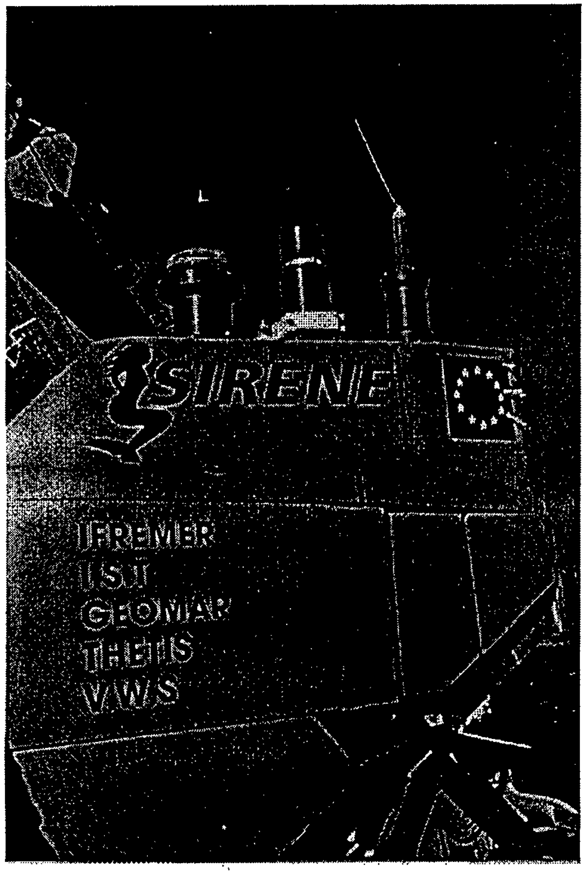
IST / Institute for Systems and Robotics
(Dynamical Systems and Ocean Robotics Lab)
Lisbon, Portugal

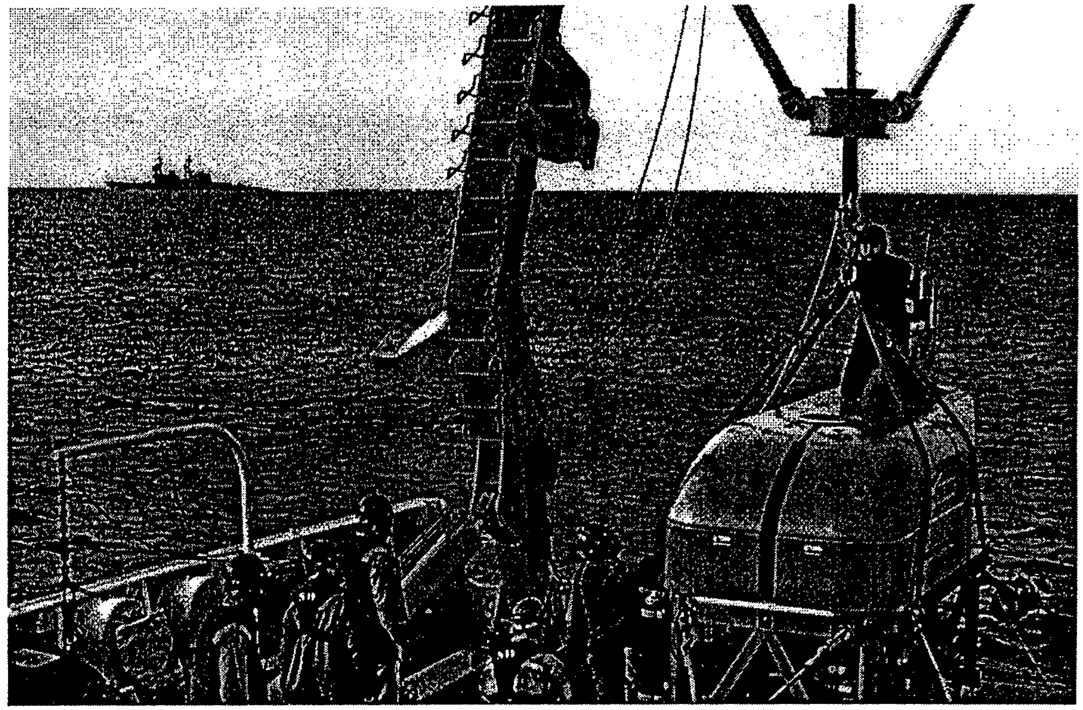
Robotic Ocean Vehicles



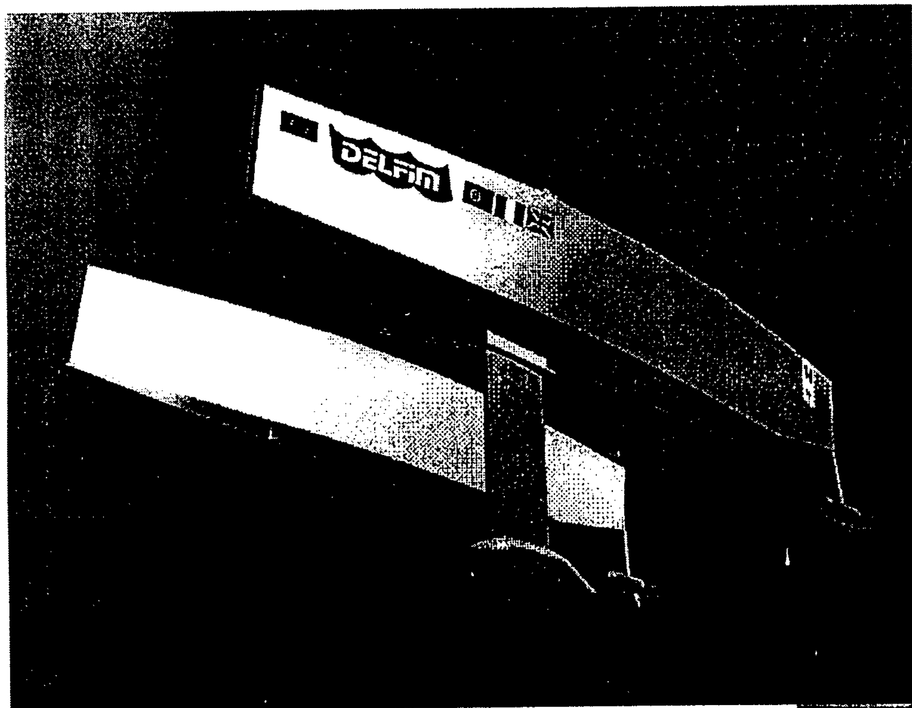


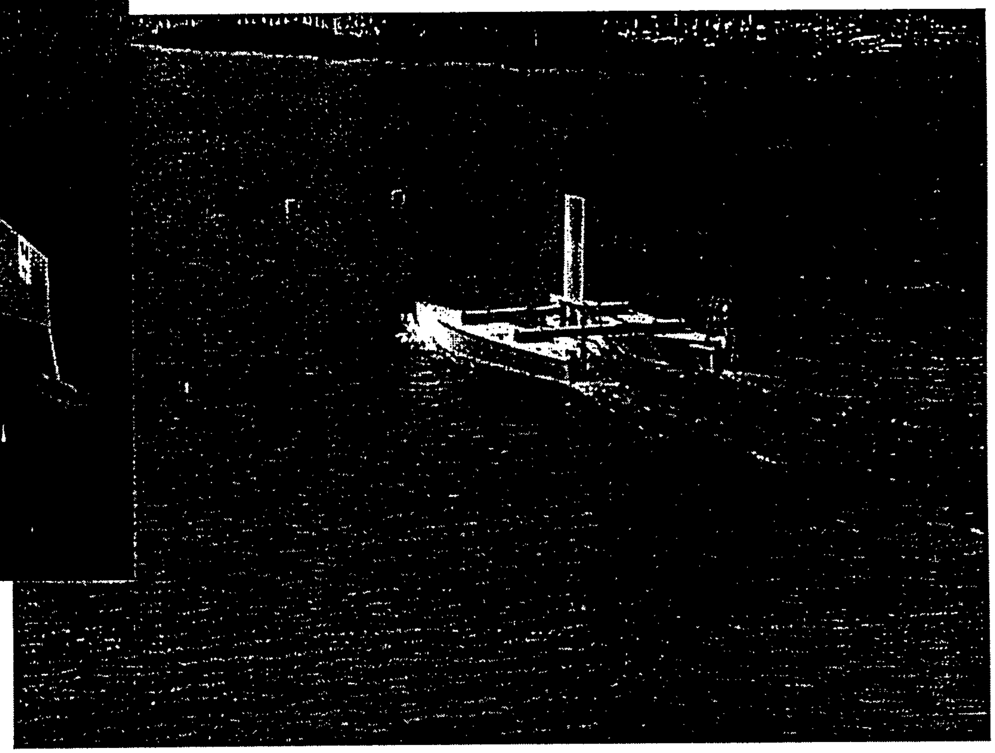
MARIUS - an Autonomous Underwater Vehicle (AUV) for Coastal Oceanography (PT, FR, DK - EC)



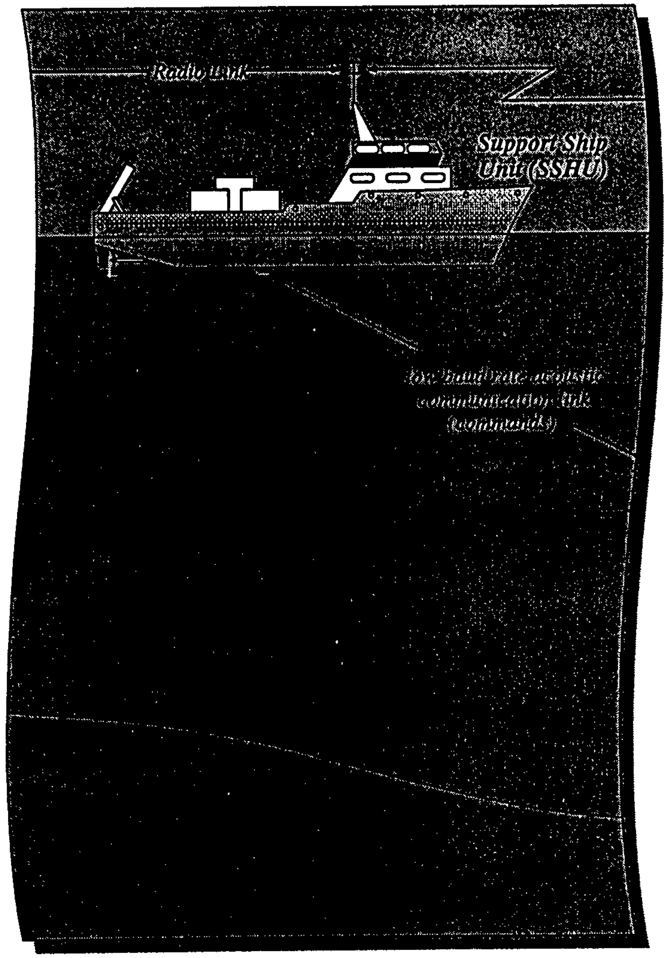


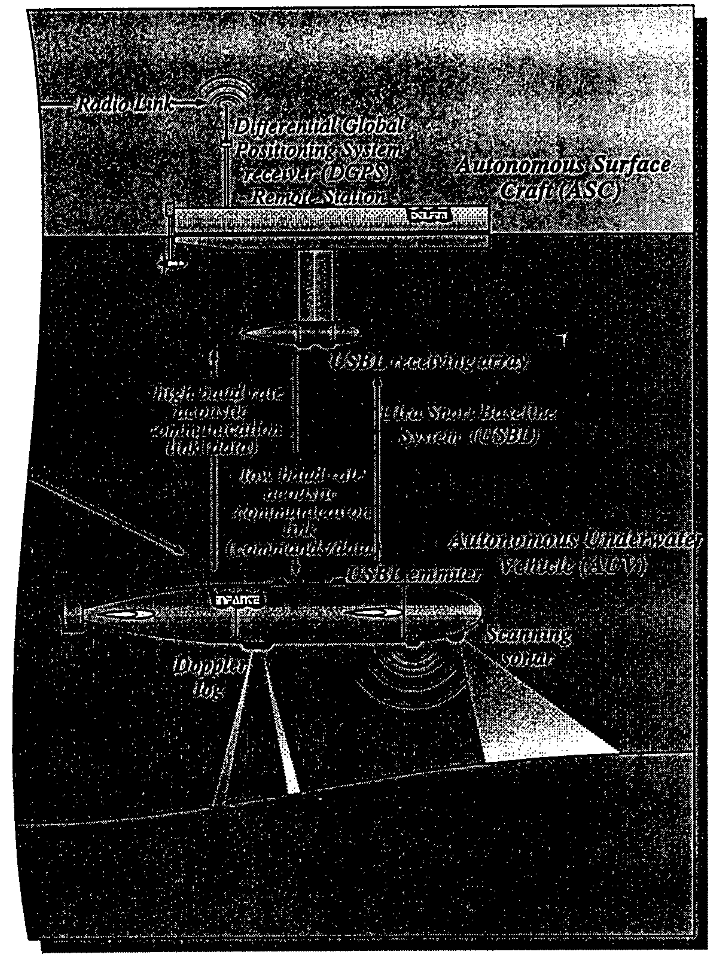
SIRENE - An Autonomous Underwater Shuttle for the Deployment of Benthic Laboratories (FR, GER, PT - EC)



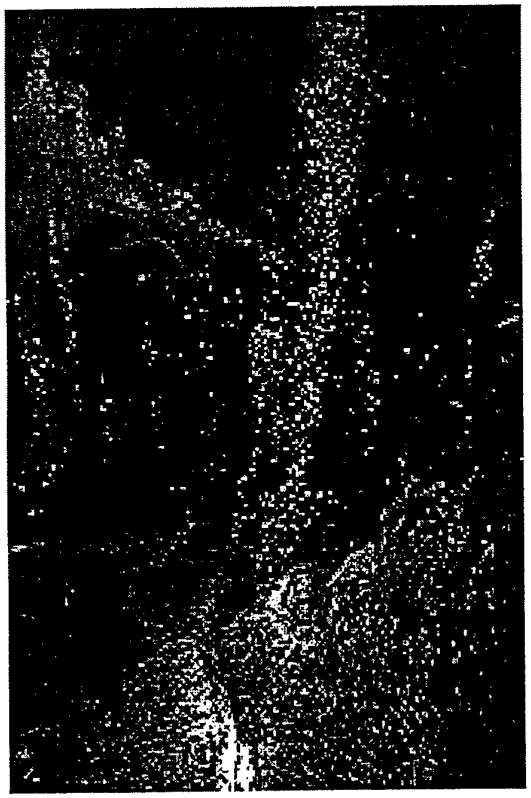


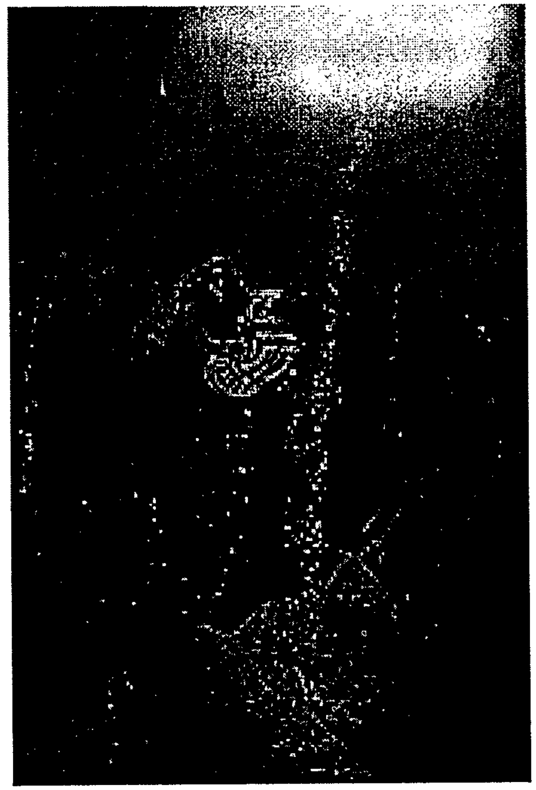
DELFIM - An Autonomous Surface Craft



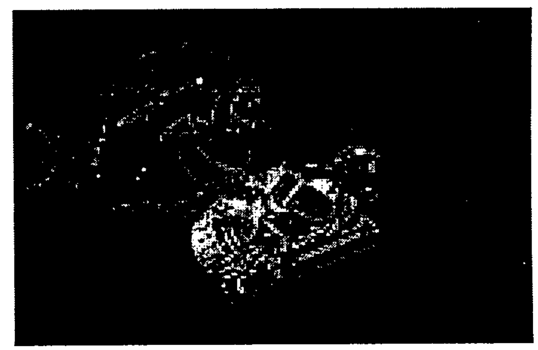


ASIMOV - Coordinated Operation of Autonomous Underwater and Surface Craft (PT, FR, UK - EC)



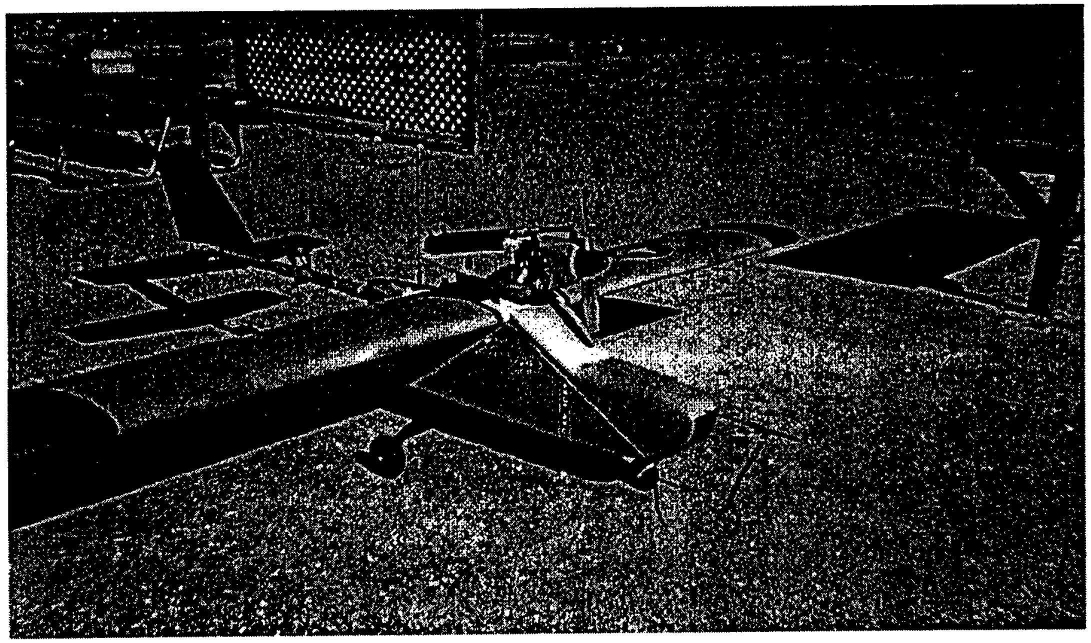


Azores, PT
Terceira Island
(first manned
survey)

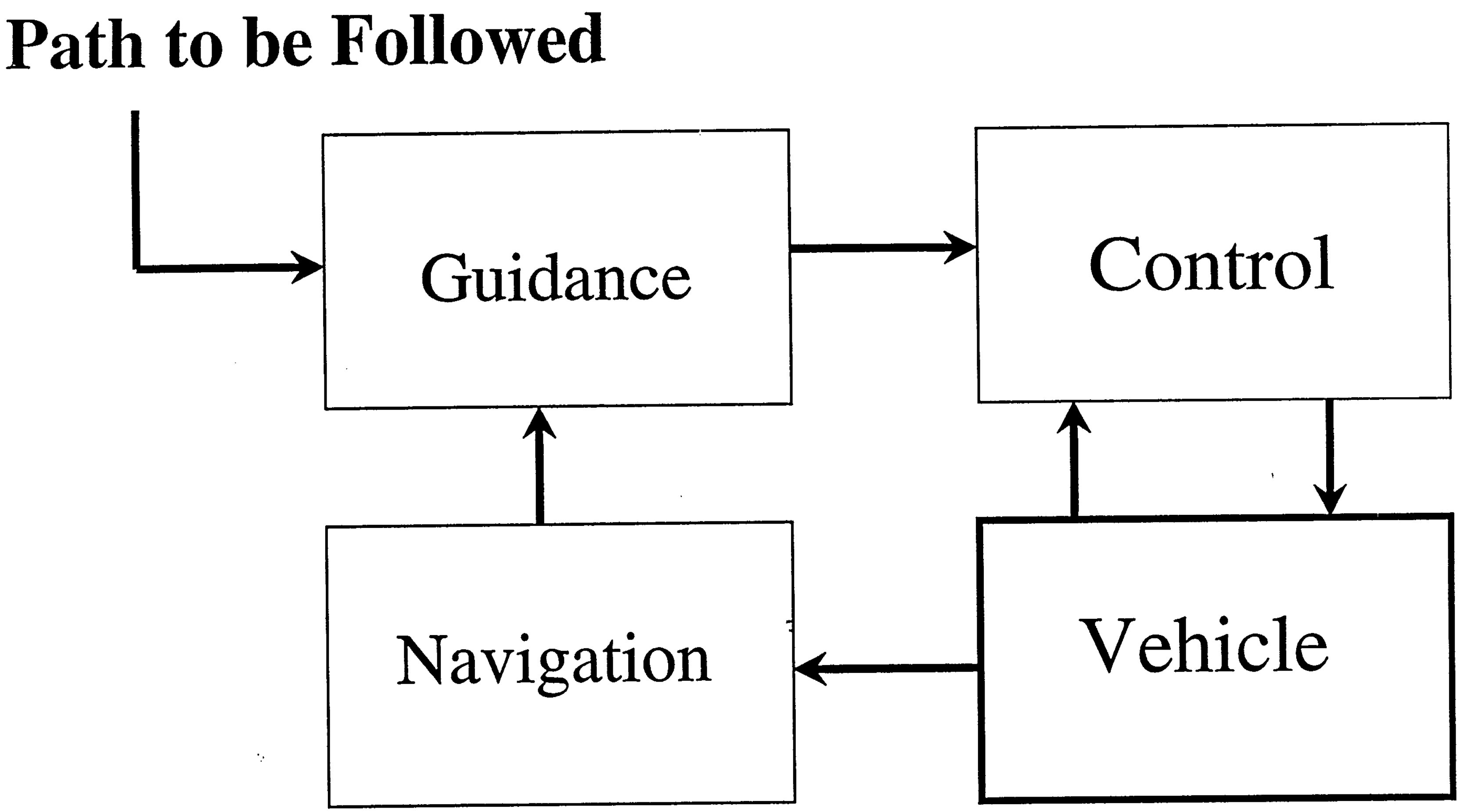


ASIMOV - Research on Hydrothermal Vent Activity.
Automatic Path Following and Video/ Sonar Data Acquisition

Robotic Air Vehicles



The FROG Unmanned Air Vehicle
(coop. with the US Naval Postgraduate School)



Basic Building Blocks of an Autonomous Vehicle

Topics Addressed

[1] - Navigation Using Time-Varying Complementary Filters (Polytopic Systems and LMIs).

[2] - Path Following for Autonomous Vehicles (Lyapunov-Based Control; Backstepping)

[3] - Combined Plant/Controller Optimization (Convex Optimization Methods).

Navigation System Design Using Time-Varying Complementary Filters

A. Pascoal[†‡] I. Kaminer[‡] P. Oliveira [†]

‡ Department of Aeronautics and Astronautics
Naval Postgraduate School
Monterey CA

† Institute for Systems and Robotics
and
Department of Electrical Engineering
Instituto Superior Tecnico
Lisboa, Portugal

Practical Problem: To develop advanced navigation systems for autonomous vehicles (air vehicles, oceanographic surface craft, underwater vehicles).

Problem Addressed: To estimate the *velocity* and *position* of an oceanographic surface craft based on measurements provided by a Doppler Log and a DGPS (differential Global Positioning System) unit.

Traditionally, time-varying navigation system design is done using Kalman-Bucy Filtering Theory.

- Requires a complete stochastic characterization of process and observation noises - a task that may be difficult, costly, or not suited to the problem at hand [Brown].
- The filters lack stability and performance guarantees
- The filter performance characterization is not compatible with conventional control system design techniques.

Alternative approach: extend well-known linear time-invariant *complementary filtering* techniques to the time-varying setting.

Main Thrust of Theoretical Research: To develop navigation system design tools that explicitly address "frequency-like" performance specifications. This allows tuning the characteristics of the navigation system to the bandwidths of the sensors used.

Theoretical Difficulty: The systems under study are time-varying.

Main Contributions:

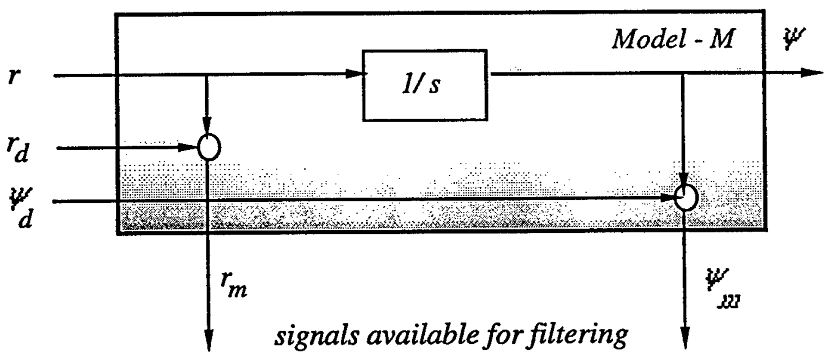
- ANALYSIS - assessment of the performance of time-varying navigation filters in a frequency setting.

- SYNTHESIS - development of a new methodology for the design of time-varying navigation systems that takes explicitly into account "frequency-like" design criteria. Theoretical tools: *linear differential inclusions* and *linear matrix inequalities* (LMIs).

- Complementary filtering: basic concepts
- Time-varying systems: mathematical background
- Navigation system design: problem formulation
- Time-varying complementary filters: main results
- Conclusions

Complementary Filters: Basic Concepts

Consider



Problem: estimate the **heading** ψ of a vehicle based on measurements r_m and ψ_m of $r = \dot{\psi}$ and ψ respectively, provided by a **rate gyro** and a **fluxgate compass**. The measurements are corrupted by **disturbances** r_d and ψ_d.

For every $k > 0$

$$\begin{aligned}\psi(s) &= \frac{s+k}{s+k}\,\psi(s) = \frac{k}{s+k}\psi(s) + \frac{s}{s+k}\,\psi(s) \\ &= T_1(s)\psi(s) + T_2(s)\psi(s),\end{aligned}$$

where $T_1(s) = k/(s+k)$ and $T_2(s) = s/(s+k)$ satisfy the equality

$$T_1(s) + T_2(s) = I.$$

Clearly,

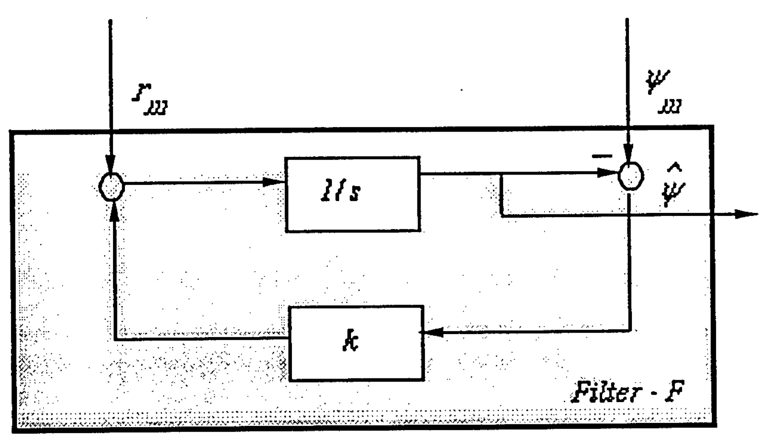
$$\psi(s) = F_\psi(s)\psi(s) + F_r(s)r(s),$$

where $F_\psi(s) = T_1(s) = k/(s+k)$ and $F_r(s) = 1/(s+k)$. This suggests a filter with the structure

$$\hat{\psi} \;=\; \mathcal{F}_\psi \psi_m + \mathcal{F}_r r_m$$

that admits the state space realization

$$\dot{\hat{\psi}} \;=\; -k\hat{\psi} + k\psi_m + r_m = r_m + k(\psi_m - \hat{\psi})$$



Simple computations show that

$$\hat{\psi} = (\mathcal{T}_1 + \mathcal{T}_2)\psi + \mathcal{F}_\psi \psi_d + \mathcal{F}_r r_d,$$

Notice the following important properties:

- $T_1(s)$ is low-pass: the filter relies on the information provided by the compass at low frequency only.
- $T_2(s) = I - T_1(s)$: the filter blends the information provided by the compass in the low frequency region with that available from the rate gyro in the complementary region.
- the break frequency is simply determined by the choice of the parameter k.

The *frequency decomposition* induced by the complementary filter structure holds the key to its practical success, since it *mimicks the natural frequency decomposition induced by the physical nature of the sensors* themselves:

- the compasses provides reliable information at low frequency only, whereas
- rate gyros exhibit biases and drift phenomena in the same frequency region and are therefore useful at higher frequencies.

Complementary Filters: Basic Concepts

Complementary filter design is reduced to the computation of the gain k to meet a target break frequency that is dictated by the physical characteristics of the sensors.

The emphasis is shifted from a stochastic framework to a deterministic framework that aims at shaping the filter closed-loop transfer functions.

Once this set-up is adopted - one is free to adopt any efficient design method (e.g.H_2 or H_∞). The design parameters are simply viewed as "tuning knobs" to shape the charateristics of the closed loop operators. Filter analysis is easily carried out in the frequency domain using Bode plots.

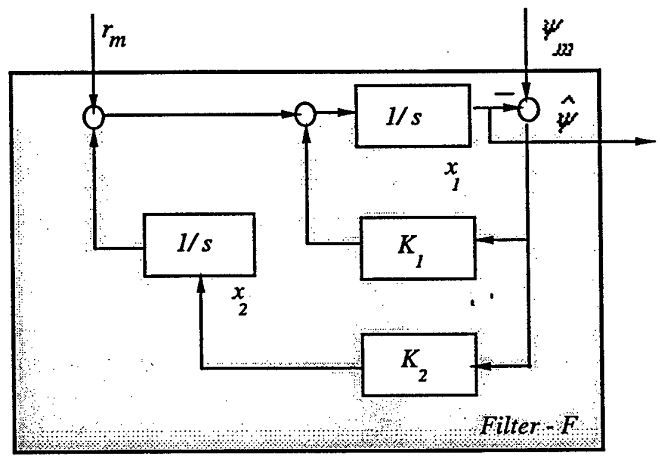
In the simple case described here, the underlying process model is

$$\begin{cases} \dot{\psi} &= r_m - r_d \\ \psi_m &= \psi + \psi_d \end{cases}$$

where r_d and ψ_d are process and measurement disturbances.

Additional requirement: design a filter to **reject the rate gyro bias**.

Solution: augment the complementary filter with an **integrator**.



Filter Realization:

$$\begin{cases} \dot{\mathbf{x}} = A\mathbf{x} + Bu + H(y - \hat{y}) \\ \hat{y} = C\mathbf{x} \end{cases}$$

where $\mathbf{x} = [x_1 x_2]^T$, $u = r_m$, $y = \psi_m$, and

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, C = \begin{bmatrix} 1 & 0 \end{bmatrix}, H = \begin{bmatrix} k_1 \\ k_2 \end{bmatrix}.$$

Simple computations show

$$\hat{\psi} = (\mathcal{T}_1 + \mathcal{T}_2)\psi + \eta$$

where

$$T_1(s) = \frac{k_1 s + k_2}{s^2 + k_1 s + k_2}, T_2(s) = \frac{s^2}{s^2 + k_1 s + k_2},$$

and

$$\eta = \mathcal{F}_\psi \psi_d + \mathcal{F}_r r_d$$

is a noise term, the intensity of which depends on

$$F_\psi(s) = T_1(s) \text{ and } F_r(s) = \frac{s}{s^2 + k_1 s + k_2}$$

Again, notice that

$$T_1(s) + T_2(s) = I$$

$T_1(s)$ is low pass, and $T_2(s)$ is high-pass.

Complementary Filters: Basic Concepts

The filter

- blends naturally the information provided by the compass at low frequency with that available from the rate gyro in the complementary frequency range
- any constant term in r_d (rate gyro bias) is rejected at the output since $F_r(0) = 0$.
- rejects naturally high frequency noise present in the fluxgate mesurements

Deterministic Framework ($r_d = \psi_d = 0$).

Definition. (r, ψ) **Complementary Filter**. Consider the process model

$$\mathcal{M}_{\psi r} := \begin{cases} \dot{\psi} & = r \\ \psi_m & = \psi \\ r_m & = r \end{cases} \tag{1}$$

and a filter $\mathcal{F}$ with realization

$$\begin{aligned} \dot{\mathrm{x}} &= A\mathrm{x} + B_r r_m + B_\psi \psi_m \\ \hat{\psi} &= C\mathrm{x} \end{aligned}$$

Then, $\mathcal{F}$ is said to be a complementary filter for $\mathcal{M}_{\psi r}$ if

- $\mathcal{F}$ it is internally stable
- For every any initial conditions $\psi(0)$ and $\mathrm{x}(0)$ $lim_{t\to\infty}\{\psi(t) - \hat{\psi}(t)\} = 0$.
- $\mathcal{F}$ satisfies a bias rejection property, that is, $lim_{t\to\infty}\hat{\psi} = 0$ when $\psi_m = 0$ and r_m is an arbitrary constant.
- The operator $\mathcal{F}_\psi : \psi_m \to \hat{\psi}$ is a finite bandwith low pass filter.

Time-varying systems: mathematical background

In preparation for what follows: need to introduce the concept of Finite Bandwith Low Pass **Time-Varying** filters.

A causal system $\mathcal{G}$ is $(finite - gain) stable$ if the *induced operator norm* (maximum energy amplification)

$$||\mathcal{G}|| := \sup\{\frac{||\mathcal{G}f||_2}{||f||_2} : f \in L_2, f \neq 0\}$$

is finite.

We will deal with linear time-varying systems with realizations

$$\{A(t), B(t), C(t), D(t)\} \in \Omega$$

$$\Omega := \mathbf{Co}\{\{A_1, B_1, C_1, D_1\}, ..., \{A_L, B_L, C_L, D_L\}\}$$

where

$$\mathbf{Co}S := \{\sum_{i=1}^{L} \lambda_i \mathcal{A}_i | \mathcal{A}_i \in S, \lambda_1 + ... + \lambda_L = 1\}$$

is the convex hull of the set $S := \{\mathcal{A}_1, ..., \mathcal{A}_n\}$. These are **polytopic differential inclusions.**

It can be shown that given a polytopic system $\mathcal{G}$, $||\mathcal{G}|| < \gamma$ if $\exists\ P > 0$ such that

$$\begin{bmatrix} A_i^T P + PA_i & PB_i & C_i^T \\ B_i^T P & -\gamma^2 I & D_i^T \\ C_i & D_i & -I \end{bmatrix} < 0; i = 1, 2, ..., L.$$

Checking that such a P exists can be done using highly efficient numerical algorithms [Boyd et al., MatLab Toolbox].

Definition. Low pass property. Let $\mathcal{G}$ be a linear, internally stable time-varying system and let $\mathcal{W}_\omega^n$ be a low-pass, linear time-invariant Chebyschev filter of order n and cutoff frequency ω. The system $\mathcal{G}$ is said to satisfy a low pass property with indices (ϵ, n) over $[0, \omega_c]$ if

$$||(\mathcal{G} - I)\, W_{\omega_c}^n|| < \epsilon$$

Definition. Low pass filter with bandwith ω_c. A linear, internally stable time-varying system $\mathcal{G}$ is said to be an (ϵ, n) *low pass filter* with bandwidth ω_c if

- $\lim_{\omega \to 0} ||(\mathcal{G} - I)\, W_\omega^n||$ is well defined and equals 0.
- $\omega_c := \sup\{\omega : ||(\mathcal{G} - I)\, W_\omega^n|| < \epsilon\}$, i.e. $\mathcal{G}$ satisfies a low pass property with indices (ϵ, n) over $[0, \omega]$ for all $\omega \in [0, \omega_c)$ but fails to satisy that property whenever $\omega \geq \omega_c$.
- For every $\delta > 0$, there exists $\omega^* = \omega^*(\delta)$ such that $||\mathcal{G}(I - W_\omega^n)|| < \delta$ for $\omega > \omega^*$.

The above conditions generalize the following facts that are obvious in the linear time-invariant case:

- the filter must provide a gain equal to one at zero frequency
- there is a finite band of frequencies over which the system behaviour replicates very closely that of an identity operator
- the system gain rolls-off to zero at high frequency.

Navigation system design: problem formulation

Basic Notation

$\{\mathcal{I}\}$ is a reference frame; $\{\mathcal{B}\}$ is a body-fixed frame that moves with the vehicle.

- $\mathbf{p} = [x\ y\ z]^T$ - position of the origin of $\{\mathcal{B}\}$ measured in $\{\mathcal{I}\}$.
- ${}^I\mathbf{v} = [\dot{x}\ \dot{y}\ \dot{z}]^T$ - linear velocity of the origin of $\{\mathcal{B}\}$ measured in $\{\mathcal{I}\}$.
- $\mathbf{v} = [u\ v\ w]^T$ - linear velocity of the origin of $\{\mathcal{B}\}$ with respect to $\{\mathcal{I}\}$, resolved in $\{\mathcal{B}\}$
- $\boldsymbol{\omega} = [p\ q\ r]^T$ - angular velocity of $\{\mathcal{B}\}$ with respect to $\{\mathcal{I}\}$, resolved in $\{\mathcal{B}\}$.
- $\boldsymbol{\lambda} = [\phi\ \theta\ \psi]^T$ - vector of roll, pitch, and yaw angles that parametrize locally the orientation of frame $\{\mathcal{B}\}$ with respect to $\{\mathcal{I}\}$.

Navigation system design: problem formulation

Rotation matrix: ${}^{I}_{B}\mathcal{R}$ (abbreviated $\mathcal{R}$) is the rotation matrix from $\{\mathcal{B}\}$ to $\{\mathcal{I}\}$, parametrized locally by $\boldsymbol{\lambda}$, that is, $\mathcal{R} = \mathcal{R}(\boldsymbol{\lambda})$.

The following kinematic relations apply

$$\dot{\mathbf{p}} = {}^{I}\mathbf{v} = \mathcal{R}\mathbf{v} \text{ and} \tag{2}$$

$$\dot{\mathcal{R}} = \mathcal{R}\mathcal{S}(\boldsymbol{\omega}), \tag{3}$$

where

$$\mathcal{S}(\boldsymbol{\omega}) := \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \tag{4}$$

is a skew symmetric matrix.

Problem: Estimate the inertial position $\mathbf{p}$ and the velocity ${}^{I}\mathbf{v}$ of an autonomous vehicle based on measurements $\mathbf{p}_m$ (available from a DGPS unit) and $\mathbf{v}_m$ (available from an onboard Doppler log) of $\mathbf{p}$ and $\mathbf{v}$, respectively.

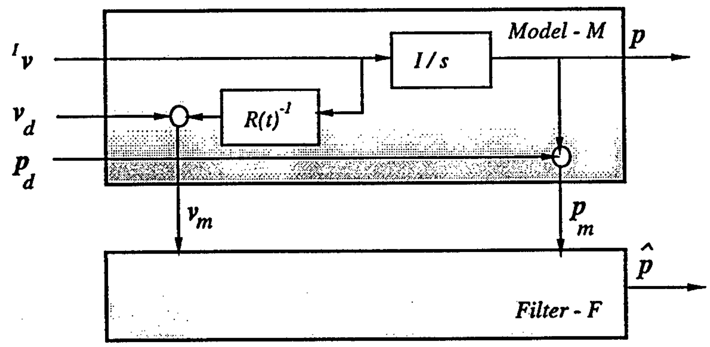
Navigation system design: problem formulation

Constraints:

- Due to the physical characteristic of the Doppler log, the measurement $\mathbf{v}_m$ is *naturally expressed in body-axis* $\{\mathcal{B}\}$. Furthermore, Doppler bias effects effects are also naturally expressed in $\{\mathcal{B}\}$.

- However, the measurements $\mathbf{p}_m$ are *directly available in the reference frame* $\{\mathcal{I}\}$.

Navigation system design: problem formulation



Definition. Process Model $\mathcal{M}_{pv}$. The **time-varying** process model $\mathcal{M}_{pv}$ is given by

$$\mathcal{M}_{pv} := \begin{cases} \dot{\mathbf{p}} &= {}^I\mathbf{v} \\ \mathbf{p}_m &= \mathbf{p} \\ \mathbf{v}_m &= \mathcal{R}^{-1}\mathbf{v} + \mathbf{v}_{d,0} \end{cases} \tag{5}$$

where $\mathbf{v}_{d,0}$ is the Doppler bias.

Assumptions: the matrix $\mathcal{R}$ and its derivative $\dot{\mathcal{R}}$ are constrained through the inequalities

$$|\phi(t)| \leq \phi_{max}, |\theta(t)| \leq \theta_{max} \tag{6}$$

and

$$|p(t)| \leq p_{max}, |q(t)| \leq q_{max}, |q(t)| \leq r_{max} \tag{7}$$

Definition. Candidate complementary filter. Consider the process model $\mathcal{M}_{pv}$ with $\mathbf{v}_{d,0}$ an arbitrary constant, and let

$$\mathcal{F} := \begin{cases} \dot{\mathbf{x}} &= A(t)\mathbf{x} + B_p(t)\mathbf{p}_m + B_v(t)\mathbf{v}_m \\ \hat{\mathbf{p}} &= C(t)\mathbf{x}. \end{cases} \tag{8}$$

Then, $\mathcal{F}$ is said to be a *candidate complementary filter* for $\mathcal{M}_{pv}$ if

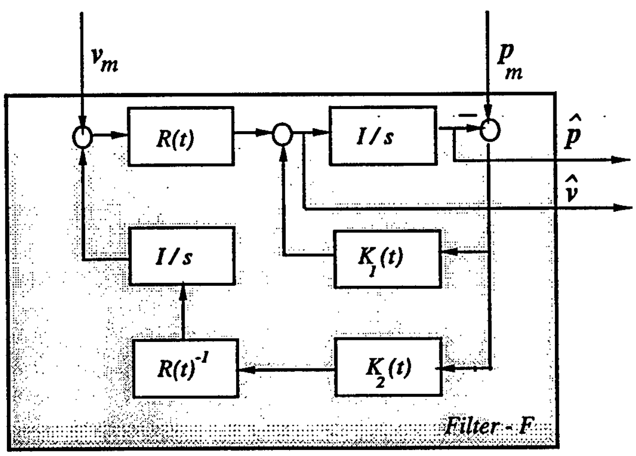
- $\mathcal{F}$ is internally stable
- For every initial conditions $\mathbf{p}(0)$ and $\mathbf{x}(0)$, $\lim_{t\to\infty}\{\psi(t) - \hat{\psi}(t)\} = 0$.
- $\mathcal{F}$ satisfies a bias rejection property, that is, $\lim_{t\to\infty} \hat{\mathbf{p}} = 0$ when $\mathbf{v} = 0$.

Definition. Complementary filter with break frequency ω_c. Let $\mathcal{F}$ be be a candidate complementary filter for $\mathcal{M}_{pv}$, and let $\mathcal{F}_p$ denote the corresponding operator from $\mathbf{p}_m$ to $\hat{\mathbf{p}}$. Then, $\mathcal{F}$ is said to be an (ϵ, n) *complementary filter* for $\mathcal{M}_{pv}$ with break frequency ω_c if $\mathcal{F}_p$ is an (ϵ, n) *low pass filter* with bandwidth ω_c.

Time-varying complementary filters: main results

Problem formulation. *Given the process model $\mathcal{M}_{pv}$ and positive numbers ω_c, n, and ϵ, find an (ϵ, n) complementary fiter for $\mathcal{M}_{pv}$ with break frequency ω_c.*

Candidate complementary filter structure



Theorem: Consider the process model $\mathcal{M}_{pv}$ and the time-varying filter

$$\mathcal{F} := \begin{cases} \dot{\mathbf{x}}_1 &= R\mathbf{v}_m + R\mathbf{x}_2 + K_1(\mathbf{p} - \mathbf{x}_1) \\ \dot{\mathbf{x}}_2 &= R^{-1}K_2(\mathbf{p} - \mathbf{x}_1) \\ \hat{\mathbf{p}} &= \mathbf{x}_1 \end{cases} \tag{9}$$

Suppose the filter $\mathcal{F}$ is internally stable. Then, $\mathcal{F}$ is a candidate complementary filter for $\mathcal{M}_{pv}$.

Time-varying complementary filters: main results

Sufficient conditions for stability and guaranteed break frequency (using *constant* gains).

Let

$$\omega_r = [p_r \; q_r \; r_r \,]^T$$

such that

$$|p_r| \leq p_r^+, |q_r| \leq q_r^+, |r_r| \leq r_r^+.$$

Then

$$\omega_r \in \mathbf{Co}\{\omega_r^i, i = \{1,..,8\}\} \text{ and}$$
$$\mathcal{S}_r \in \mathbf{Co}\{\mathcal{S}(\omega_r^i), i = \{1,..,8\}\}$$

where

$$\omega_r^1 = \begin{bmatrix} p_r^- \\ q_r^- \\ r_r^- \end{bmatrix}, \omega_r^2 = \begin{bmatrix} p_r^+ \\ q_r^- \\ r_r^- \end{bmatrix}, \omega_r^3 = \begin{bmatrix} p_r^- \\ q_r^+ \\ r_r^- \end{bmatrix}, \omega_r^4 = \begin{bmatrix} p_r^+ \\ q_r^+ \\ r_r^- \end{bmatrix}, ...\omega_r^8 = \begin{bmatrix} p_r^+ \\ q_r^+ \\ r_r^+ \end{bmatrix}.$$

$$p_r^- = -p_r^+, q_r^- = -q_r^+, r_r^- = -r_r^+$$

Time-varying complementary filters: main results

Sufficient conditions for stability and guaranteed break frequency

Theorem Consider the linear time-varying filter (9) and assume that the bounds on $\boldsymbol{\omega}_r$ apply. Given n and ω_c, let

$$\mathcal{W}^n_{\omega_c} := \left[\begin{array}{c|c} A_W & B_W \\ \hline C_W & 0 \end{array}\right]$$

be a minimal realization for a weighting Chebyschev filter Further let

$$F = \begin{bmatrix} 0 & I \\ 0 & \mathcal{S}_r \end{bmatrix}, \quad H = [-I \;\; 0] .$$

Time-varying complementary filters: main results

Sufficient conditions for stability and guaranteed break frequency

Suppose that given $\epsilon > 0 \; \exists \; M \in \mathcal{R}^{6\times 3}$, $P_1 \in \mathcal{R}^{6\times 6}, P_2 \in \mathcal{R}^{6\times 6}$, $P_1 > 0$, $P_2 > 0$ such that the *linear matrix inequalities*

$$\begin{bmatrix} F_i^T P_1 + H^T M^T + P_1 F_i + MH + H^T H & MC_W + H^T C_W & 0 \\ (MC_W + H^T C_W)^T & P_2 A + A^T P_2 + C_W^T C_W & P_2 B_W \\ 0 & B_W^T P_2 & -\epsilon^2 I \end{bmatrix} < 0,$$

$$F_i = \begin{bmatrix} 0 & I \\ 0 & \mathcal{S}(\omega_r^i) \end{bmatrix}, \quad i = \{1, \ldots 8\} \tag{10}$$

are satisfied. Then, the constant gains

$$\begin{bmatrix} K_1 \\ K_2 \end{bmatrix} = P_1^{-1} M$$

make the filter $\mathcal{F}$ internally stable. Furthermore, the operator $\mathcal{F}_p : \mathbf{p} \to \hat{\mathbf{p}}$ satisfies a low pass property with indices (ϵ, n) over $[0, \omega_c]$, that is, $||(\mathcal{F}_p - I)\, W_{\omega_c}^n|| < \epsilon$.

Therefore, the problem is reduced to checking the feasibility of the set of linear matrix inequalities

Indication of Proof

Given the candidate filter $\mathcal{F}_p$, consider the Lyapunov coordinate transformation

$$\zeta(t) = \bar{P}(t)\mathbf{x}(t),$$

where

$$\bar{P}(t) = \begin{bmatrix} I & 0 \\ 0 & R(t) \end{bmatrix}.$$

With this change of coordinates, the operator $\mathcal{F}_p$ admits the realization

$$\mathcal{F}_p = \begin{cases} \dot{\zeta} &= (\bar{P}A\bar{P}^{-1} + \dot{\bar{P}}\bar{P}^{-1})\zeta + \bar{P}B_p\mathbf{p} \\ \hat{\mathbf{p}} &= C\bar{P}^{-1}\zeta \end{cases} \tag{11}$$

Using the relations

$$\bar{P}A\bar{P}^{-1} = \begin{bmatrix} -K_1 & I \\ -K_2 & 0 \end{bmatrix}$$

and

$$\dot{\bar{P}}\bar{P}^{-1} = \begin{bmatrix} 0 & 0 \\ 0 & \mathcal{R}\mathcal{S}(\omega)\mathcal{R}^{-1} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \mathcal{S}(\mathcal{R}\omega) \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & \mathcal{S}(\omega_r) \end{bmatrix}$$

the realization can be written as

$$\begin{aligned}\dot{\boldsymbol{\zeta}} &= \begin{bmatrix} -K_1 & I \\ -K_2 & \mathcal{S}(\boldsymbol{\omega}_r) \end{bmatrix} \boldsymbol{\zeta} + \begin{bmatrix} K_1 \\ K_2 \end{bmatrix} \mathbf{p} \\ \hat{\mathbf{p}} &= [I \;\; 0]\, \boldsymbol{\zeta} \end{aligned} \tag{12}$$

Simple algebra now shows that $(\mathcal{F}_p - I)\, W^n_{\omega_c}$ admits the state-space representation

$$(\mathcal{F}_p - I)\, W^n_{\omega_c} := \left[\begin{array}{ccc|c} -K_1 & I & K_1 C_W & 0 \\ -K_2 & \mathcal{S}_r & K_2 C_W & 0 \\ 0 & 0 & A_W & B_W \\ \hline I & 0 & -C_W & 0 \end{array}\right] \tag{13}$$

$$= \left[\begin{array}{cc|c} F + KH & KC_W & 0 \\ 0 & A_W & B_W \\ \hline H & -C_W & 0 \end{array}\right]$$

$$\in \mathbf{Co} \left\{ \left[\begin{array}{cc|c} F_i + KH & KC_W & 0 \\ 0 & A_W & B_W \\ \hline H & -C_W & 0 \end{array}\right], i = \{1, ..., 8\} \right\}.$$

where

$$K = \begin{bmatrix} K_1 \\ K_2 \end{bmatrix}$$

and F, H, and F_i are defined as before. The Theorem follows from the computation of the induced operator norm of the polytopic system $(\mathcal{F}_p - I)\, W^n_{\omega_c}$.

A Practical Algorithm for Navigation System Design

Mathematical tools were introduced to design a candidate complementary filter with a *guaranteed break frequency.*

Notice: the outcome of the design process may very well be a filter with an effective bandwidth that is greater than the one required.

The set of possible solutions must be further constrained so that the designer have an extra design parameter at his disposal to select one solution (if it exists) that meets the required bandwidth criterion.

Solution in the **linear time-invariant case: make the filter "roll-off sufficiently early in frequency".**

A Practical Algorithm for Navigation System Design

In the **time-varying** setting: make $||\mathcal{F}_p(I - \mathcal{W}_{\omega_t}^{n_t})||$ sufficiently small for adequate choices of ω_t and n_t, which play the role of "tuning parameters".

Practical algorithm: **modify the algorithm described to include a new "high-frequency" constraint, which can be easily cast as a Linear Matrix Inequality.**

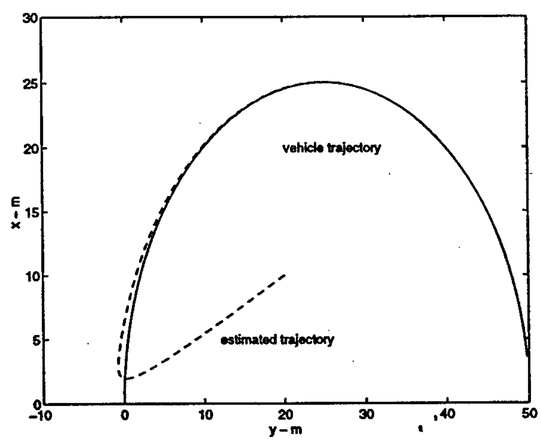
It is up to the system designer to select appropriate values of the tuning parameters to try and meet all the criteria that are required for the complementary filter.

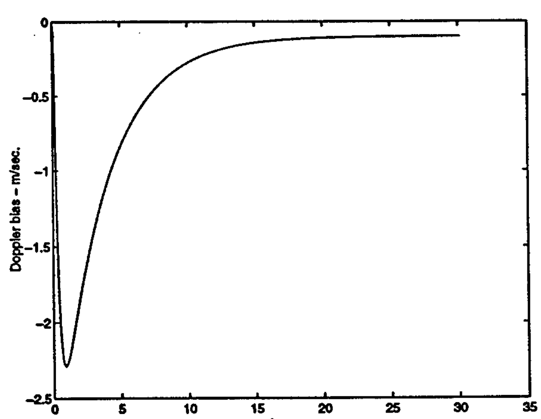
Example

– Vehicle progresses at $2m/s$ with maximum yaw rate $3rad/s$.

– The Doppler log has a bias $\mathbf{v}_{d,0} = [0.1\ m/s,\ 0.2\ m/s]^T$.

– The selected break frequency was $\omega_c = 0.1rad/s$.





- A new methodology was developed to **design linear time-varying complementary filters in a frequency setting.**

- The problem of filter design was cast in the framework of linear differential inclusions (polytopic systems).

- The design method involves determining the feasibility of a set of linear matrix inequalties

- FUTURE WORK: extend the results to the discrete-time, multi-rate case by exploring the well known isomorphism betwee multi-rate and invariant systems.

Integrated Design of Guidance and Control Systems for AUVs

P. Encarnação A. Pascoal

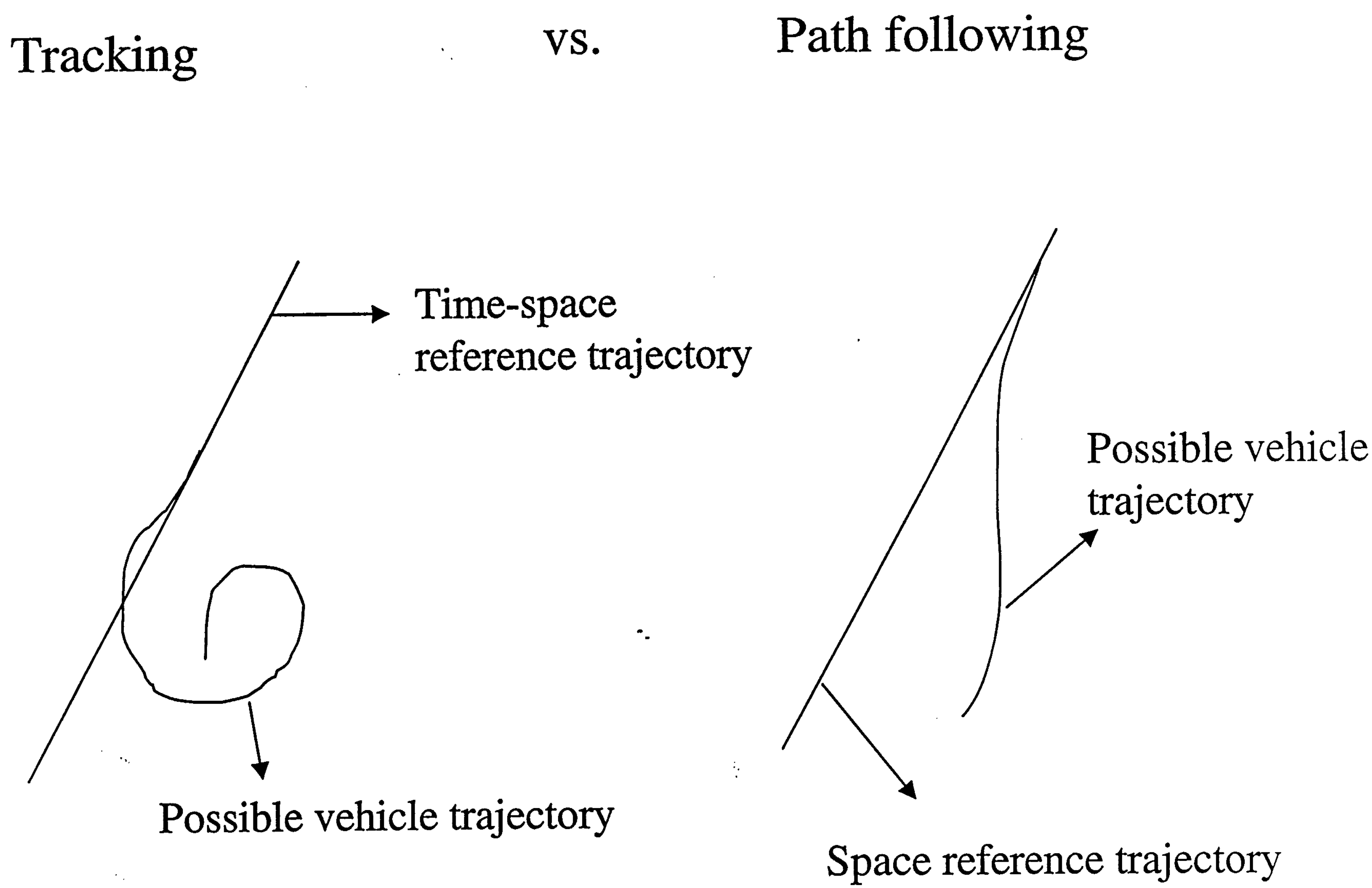
Institute for Systems and Robotics
Instituto Superior Técnico
Torre Norte, Av. Rovisco Pais, 1
1049-001 Lisboa, Portugal

Integrated Design of Guidance and Control Systems for AUVs

Outline

- Motivation and previous work review
- Integrated design description
  - Error space
  - ‘Kinematic’ controller
  - Backstepping kinematics into dynamics
  - Simulation results
- Open problems

Traditional	vs.	**Integrated design**
•Simple strategies for guidance (ex: line- of-sight) and well established design methods for control		•More involved design methods
•No stability or performance guarantees for the two systems combined		•Stability and performance specifications addressed directly.


Tracking
vs.
Path following
Time-space
reference trajectory
Possible vehicle trajectory
Possible vehicle
trajectory
Space reference trajectory

Previous work

- Several papers by **Claude Samson** (INRIA, France) *et al* on tracking and path following for land vehicles. The methods are developed for the two dimensional space and rely on vehicles kinematics only.

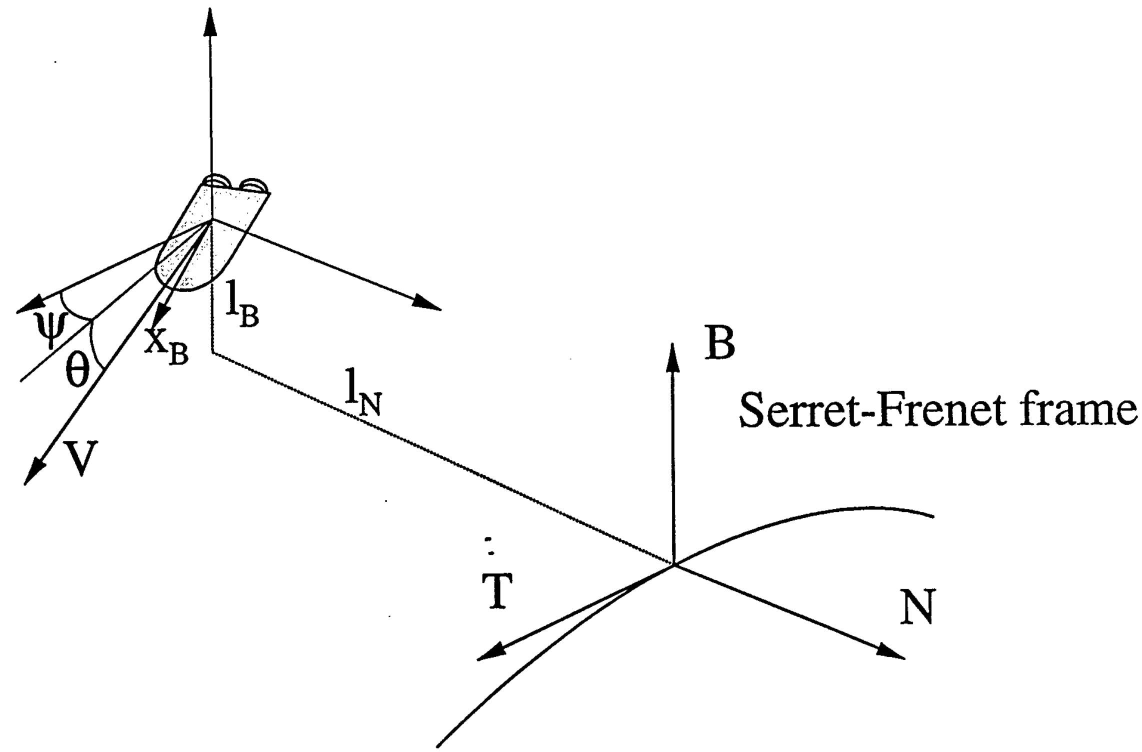
- Some papers by **Dale Enns** (Honeywell and Univ. of Minnesota, USA) *et al.* using (algebraic and numeric) dynamic inversion methods.

- A paper by **David Boyle** (Univ. of Sydney, Australia) using an iterative method for trimming, providing the references for a feed forward loop.

- **C. Silvestre, A. Pascoal** (IST, Portugal) and I. **Kaminer** (NPS, USA) used the time-invariance property of linearizations about trimming trajectories to design linear gain scheduled controllers.

- **O. Egeland** (Norway), **R. Hindman** (Univ. of Colorado, USA), ...

Integrated design: error space



Kinematic equations in the Serret-Frenet frame

Propagating the body linear velocities to the Serret-Frenet frame and computing the relative angular velocities between the wind and Serret-Frenet frames, gives

$$\begin{cases} {}^{S}R_{W}\,{}^{W}v_{B} = {}^{S}v_{S} + \dfrac{d\,{}^{S}P_{org_W}}{dt} + {}^{S}\omega_{S} \times {}^{S}P_{org_W} \\ {}^{W}\omega_{W} = {}^{W}R_{B}\,{}^{B}\omega_{B} - {}^{W}R_{s}\,{}^{S}\omega_{S} \end{cases}$$

Kinematic equations

$$
\begin{cases}
\dot{s} = \dfrac{V\cos\psi\cos\theta}{1 - l_N k} \\
\dot{l}_N = V\sin\psi\cos\theta + l_B\tau\dot{s} \\
\dot{l}_B = -V\sin\theta - l_N\tau\dot{s} \\
\dot{\phi} = p' - \tau\cos\psi\cos\theta\dot{s} + \sin\phi\tan\theta q' - \tau\tan\theta\cos\psi\sin\theta\dot{s} + \cos\phi\tan\theta r' \\
\dot{\theta} = \cos\phi q' + \tau\sin\psi\dot{s} - \sin\phi r' \\
\dot{\psi} = \dfrac{\sin\phi}{\cos\theta} - \tau\tan\theta\cos\psi\dot{s} - k\dot{s} + \dfrac{\cos\phi}{\cos\theta}r'
\end{cases}
$$

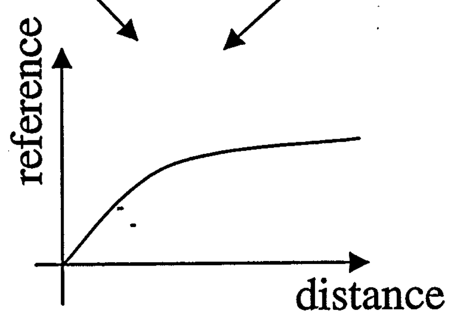
where $\begin{bmatrix} p' \\ q' \\ r' \end{bmatrix} = {}^{W}R_B \begin{bmatrix} p \\ q \\ r \end{bmatrix}$ are the body angular velocities expressed in the wind axis.

Note: for path following purposes, the total velocity is considered constant.

Lyapunov based control design

tradeoff between the competing goals of driving the orientation and distance errors to zero

$$V = \frac{1}{2}\left(l_N^2 + l_B^2 + a(\psi - f(l_N))^2 + b(\theta - g(l_B))^2\right)$$



• As the vehicle has no actuation on p and an error on the rotation about the x_W axis does not compromise path following, ϕ is left free.

Control law

accounting for the roll angle

control law for a vehicle with $\phi=\alpha=\beta=0$

$$\begin{bmatrix} q \\ r \end{bmatrix} = \begin{bmatrix} \frac{1}{c\beta} & t\alpha t\beta \\ 0 & \frac{1}{c\alpha} \end{bmatrix} \begin{bmatrix} c\phi & s\phi \\ -s\phi & c\phi \end{bmatrix} \begin{bmatrix} \frac{l_B V}{a}\frac{s\theta - sf}{\theta - f} - \tau s\psi \dot{s} - f'_{l_B} V s\theta - f'_{l_B} l_N \tau \dot{s} - k_\theta(\theta - f) \\ -\frac{l_N V c^2\theta}{b}\frac{s\psi - sg}{\psi - g} + c\theta(\tau c\psi t\theta + k)\dot{s} + g'_{l_N} V s\psi c^2\theta + g'_{l_N} l_B \tau \dot{s} c\theta - k_\psi(\psi - g) \end{bmatrix}$$

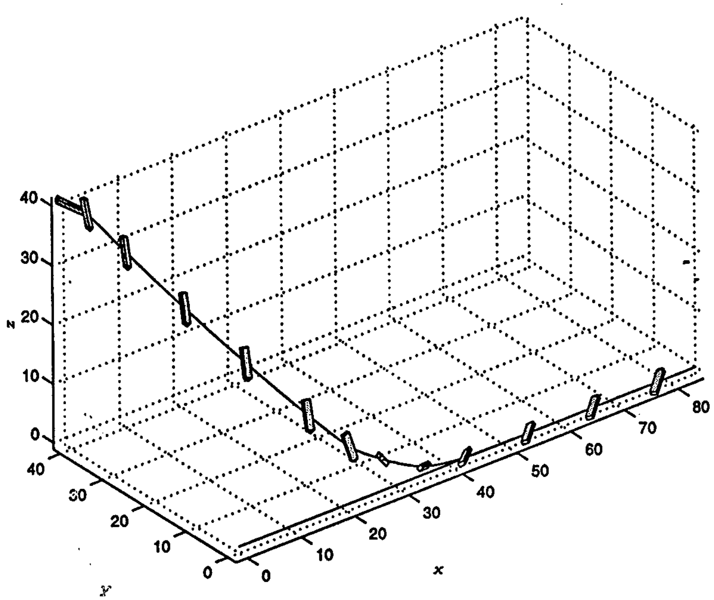
$$+ \begin{bmatrix} \frac{t\beta}{c\alpha} p \\ t\alpha p \end{bmatrix}$$

from the wind axis back to the body axis

Note: the angles α, and β, and the angular velocity p will be set by the vehicle dynamics.

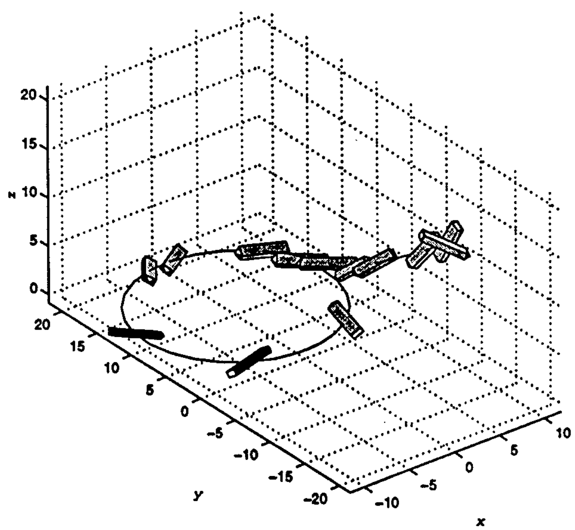
Simulation results

In the following simulations, the values of α, β, and p are artificial and only intend to show the convergence of the control system independently of those ‘dynamic’ variables.

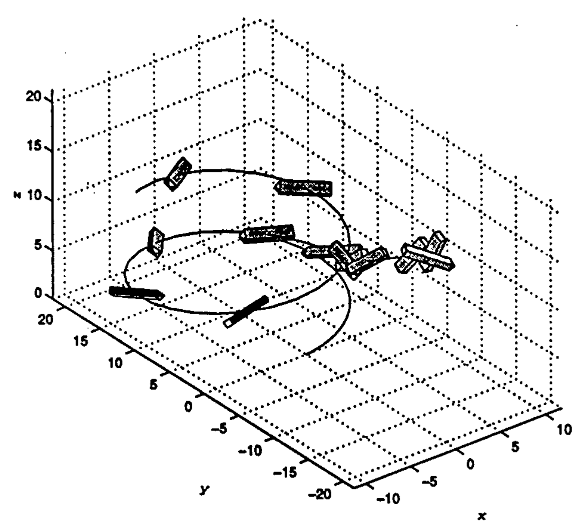


Vehicle following a straight line with $\alpha=\beta=30°$ and $p=0.01$rad/s

Simulation results

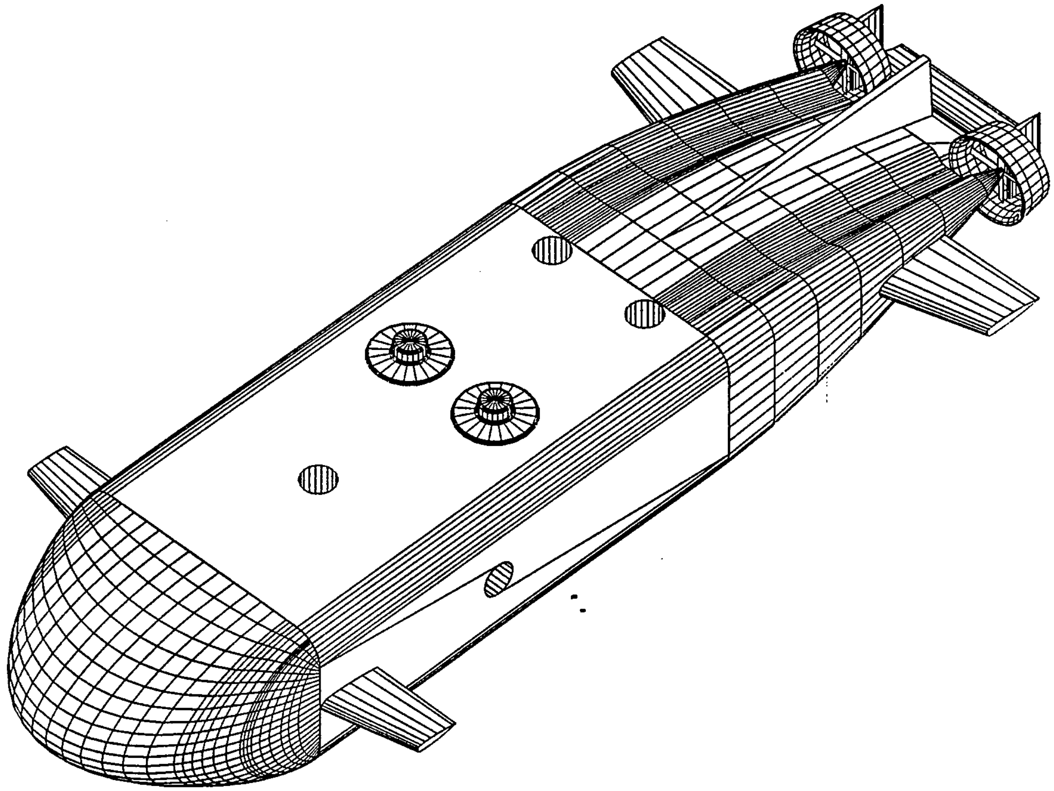


Vehicle following a circumference with $\alpha=\beta=30°$ and $p=0.01$rad/s



Vehicle following a helix with $\alpha=\beta=30°$ and $p=0.01$rad/s

The INFANTE vehicle



Note: the bow and stern control surfaces for the vertical plane are combined for this case study into an equivalent control surface for which each surface contribution is proportional to its hydrodynamic efficiency.

Vehicle dynamics

the propulsion force is chosen so that V is held constant.

$$
\begin{cases}
\dot{V} = \frac{1}{m}(X\cos\alpha\cos\beta + Y\sin\beta + Z\sin\alpha\cos\beta) \\
\dot{\alpha} = q + \frac{1}{V\cos\beta}\left(-\frac{1}{m}X\sin\alpha + \frac{Z}{m}\cos\alpha - Vp\sin\beta\cos\alpha - Vr\sin\beta\sin\alpha\right) \\
\dot{\beta} = p\sin\alpha - r\cos\alpha + \frac{1}{mV}(-X\sin\beta\cos\alpha + Y\cos\beta - Z\sin\beta\sin\alpha) \\
\dot{p} = \frac{I_y - I_z}{I_x}qr + \frac{K}{I_x} \\
\dot{q} = \frac{I_z - I_x}{I_y}pr + \frac{M}{I_y} \\
\dot{p} = \frac{I_x - I_y}{I_z}pq + \frac{N}{I_z}
\end{cases}
$$

$[X \quad Y \quad Z \quad K \quad M \quad N]$ is the vector of external forces and moments, including hydrodynamic and restoring forces and moments, and propulsion force.

Backstepping kinematics into dynamics - first step

- Looking at the equations for q and r, one can see that the system as a vector relative degree of 1;

- Using a dynamic inversion control law, one gets the following decoupled system

$$\begin{cases} \dot{q} = \upsilon_q \\ \dot{r} = \upsilon_r \end{cases}$$

Backstepping kinematics into dynamics - second step

Kinematic and dynamic vehicle model in compact form

State vector $\eta = [l_N \quad l_B \quad \theta \quad \psi]^T$

Virtual input vector $\xi = [q \quad r]^T$

$$\upsilon = [\upsilon_q \quad \upsilon_r]^T$$

$$\begin{cases} \dot{\eta} = \mathbf{f}(\eta, \xi) + \mathbf{g}(\eta, \xi)\xi & \text{Kinematic equations} \\ \dot{\xi} = \upsilon & \text{Dynamic equations} \end{cases}$$

Backstepping kinematics into dynamics - second step

•Setting ξ as in the 'kinematic' controller and using the Lyapunov function

$$V_1 = \frac{1}{2}\left(l_N^2 + l_B^2 + a(\psi - f(l_N))^2 + b(\theta - g(l_B))^2\right)$$

one can prove that $\eta \to 0$.

•With $z = \xi - \xi_d$ the system equations become

$$\begin{cases} \dot{\eta} = \mathbf{f}(\eta,\xi) + \mathbf{g}(\eta,\xi)\xi_d + \mathbf{g}(\eta,\xi)z \\ \dot{z} = \upsilon - \dot{\xi}_d \end{cases}$$

Backstepping kinematics into dynamics - second step

Lyapunov design for the complete system

$$V_2 = V_1 + \frac{1}{2} z z^T$$

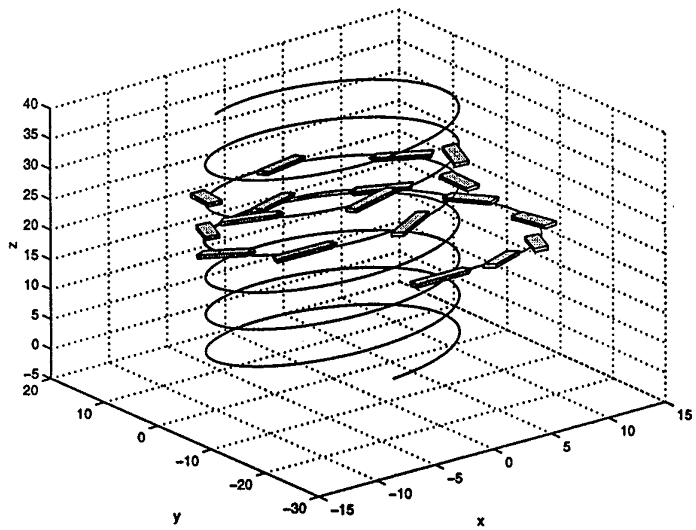
ξ should converge for the desired valued

Control law

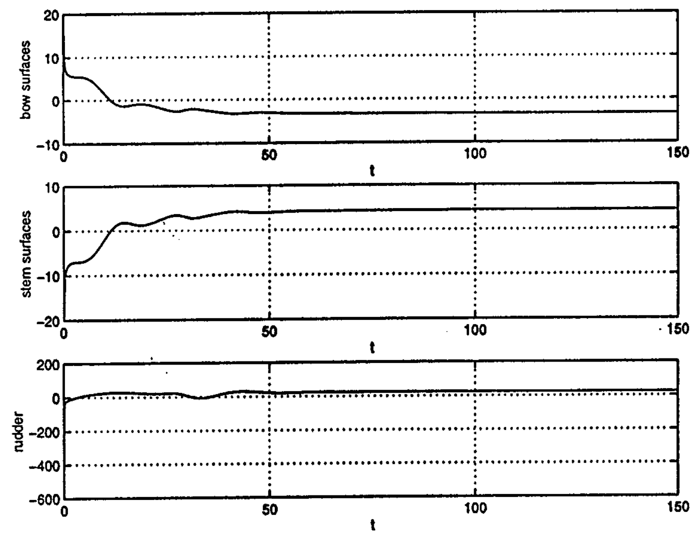
$$\upsilon = \dot{\xi}_d - \left(\frac{\partial V_1}{\partial \eta} g \right)^T - Kz$$

definite positive gain matrix

Simulation results



Vehicle following an helix



Control signals (surface deflections)

Open problems

- Incorporate the actuator dynamics (including saturation limits) directly in the design process.

- Reject constant perturbations (sea currents);

- Apply the results to unmanned air vehicles.

Combined Plant / Controller Optimization with Applications to Autonomous Underwater Vehicles (AUVs)

C. Silvestre †, A. Pascoal †‡*, I. Kaminer ‡, A. Healey *

† Instituto Superior Técnico

Institute for Systems and Robotics - Portugal

‡ Department of Aeronautics and Astronautics

Naval Postgraduate School, Monterey, CA 93943, USA

* Autonomous Underwater Vehicles Laboratory

Department of Mechanical Engineering

Naval Postgraduate School, Monterey, CA 93943

Plant Controller Optimization Problem (PCO)

Given an AUV - with a fixed baseline body configuration - that is required to operate over a finite set of representative trimming conditions in the vertical plane, determine the optimal size of the bow and stern control surfaces so that a weighted average of the power required at the trimming conditions is minimized, subject to the conditions that:

i) open loop requirements are met and

ii) stabilizing feedback controllers can be designed to meet time and frequency closed loop performance requirements about each trimming point.

Open loop requirements.

- Possibility of achieving trim at each operating condition.
- Meeting a desired degree of open-loop stability.

Closed loop requirements.

- Maneuverability specifications in response to depth commands.
- Hard limits on surface deflections.
- Actuator bandwidth constraints.

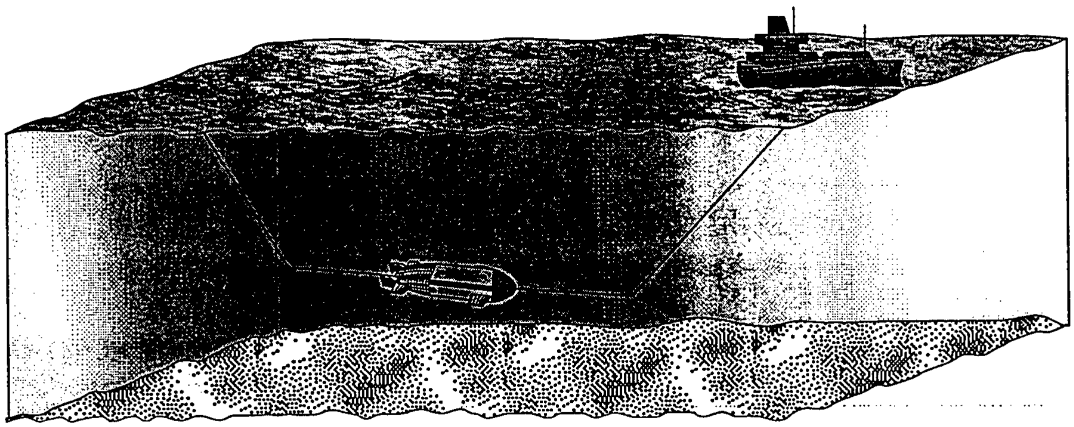


Figure 1: A typical three phase vehicle mission.

The 50 minute mission has three distinct phases:

- phase 1, duration 15 minutes: dive to 233 m depth with velocity $v_{t_0} = 1.0\,\mathrm{m/s}$ and flight path angle $\gamma_0 = -15$ deg.

- phase 2, duration 20 minutes: perform a survey over a 3000 m stretch above the sea bed while on straight level flight at cruise speed 2.5 m/s h

- phase 3, duration 15 minutes: re-surface with velocity of $v_{t_0} = 1.0\,\mathrm{m/s}$ at flight path angle $\gamma_0 = -15$ deg.

Key idea. Cast the PCO problem in the form of a new *constrained optimization problem* where

- the cost J to be minimized is the average power required at trimming.
- the search is done over the set of feedback controllers that meet open loop and closed loop requirements.

Facts The cost J can be written explicitly in terms of the vehicle surface sizes. The open and closed loop requirements can be expressed as Linear Matrix Inequalities (LMIs) that are also functions of the control surface sizes.

Optimization problem The PCO problem is reduced to minimizing a certain function of the surface sizes, while satisfying a finite set of LMI constraints.

Tools The new problem is solved numerically by resorting to efficient convex optimization algorithms / LMI Toolbox.

Basic Notation.

$\{B\}$ - body fixed frame; $\{I\}$ - reference frame.

$\mathbf{p} = [x, z]'$ - position of the origin of $\{B\}$ measured in $\{I\}$;

$\mathbf{v} = [u, w]'$ - body-fixed linear velocity;

θ - pitch angle;

q - angular velocity of $\{B\}$ relative to $\{I\}$;

$\dot{\mathbf{q}}_v' = [u, w, q]'$ - extended velocity vector in the vertical plane;

$\boldsymbol{\delta} := [\delta_b, \delta_s]'$ bow and stern plane deflections.

Underwater vehicle model.

$$M_{RB_v}\ddot{\mathbf{q}}_v + C_{RB_v}(\dot{\mathbf{q}}_v)\dot{\mathbf{q}}_v = \boldsymbol{\tau}_v(\ddot{\mathbf{q}}_v, \dot{\mathbf{q}}_v, \theta, \boldsymbol{\delta}, T)$$

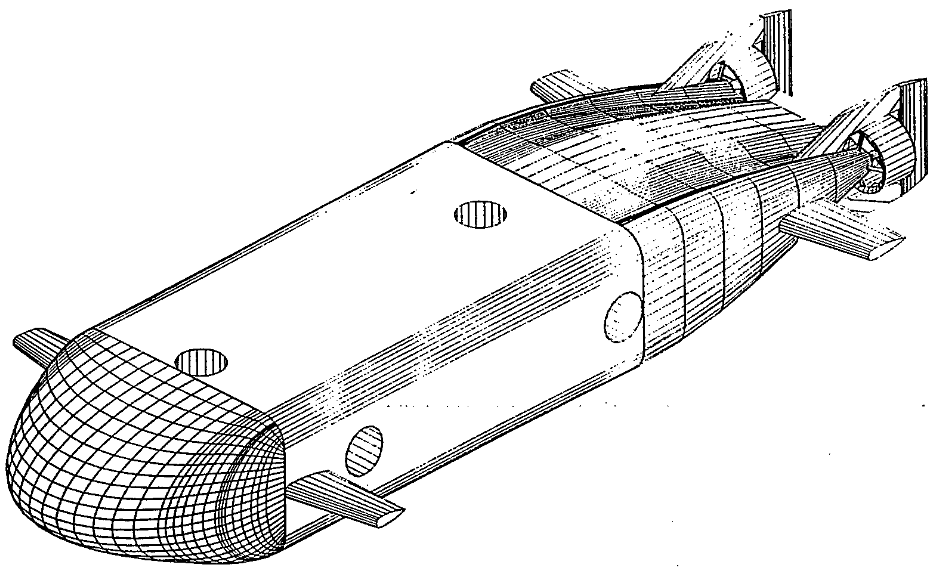
$$\dot{\theta} = q; \quad \dot{z} = -usin\theta + wcos\theta$$

where

$\boldsymbol{\tau}_v$ - vector of external forces and moments,

M_{RB_v} - generalized inertia matrix,

C_{RB_v} - matrix of Coriolis and centripetal terms,



Parameterization of the AUV vertical plane equations of motion in terms of the *bow* and *stern* control surface sizes ζ_b and ζ_s.

Assumptions

- the cord c and length d of the control surfaces are such that their aspect ratio $AR = d/c$ is constant.
- the control surfaces have a constant profile.
- thrpghout the optimization procedure the control surface rotation axes are maintained at fixed positions.

Trim (equilibrium) point: set of input and state variables for which the net sum of the forces and moments acting on the vehicle is zero.

Equilibrium point - formal definition: a vector

$$(\dot{\mathbf{q}}_{v0}, \delta_{b0}, \delta_{s0}, \theta_0, T_0)$$

such that

$$C_{RB_v}(\dot{\mathbf{q}}_{v0})\dot{\mathbf{q}}_{v0} - \boldsymbol{\tau}_v(0, \dot{\mathbf{q}}_{v0}, \theta_0, \boldsymbol{\delta}_0, T_0) = 0$$

The only equilibrium points of the AUV in the vertical plane are those that correspond to *straight line trajectories*, parameterized in terms of *total speed* v_t and *flight path angle* γ.

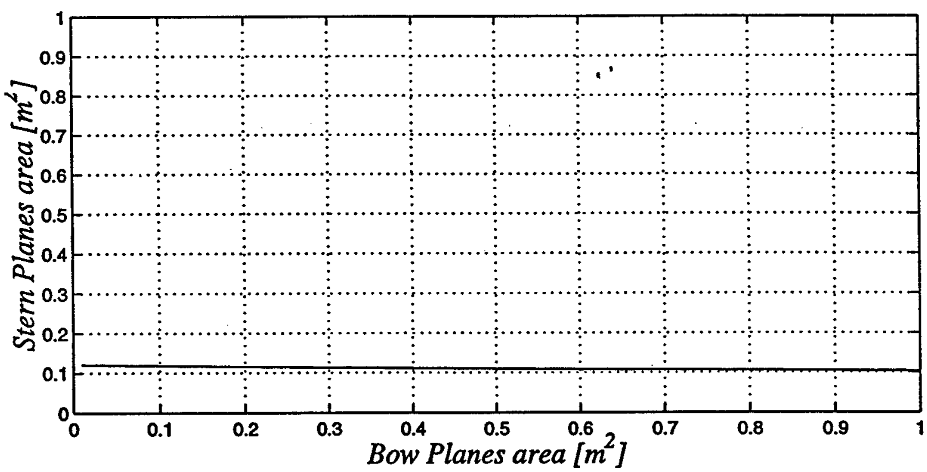
Due to the existence of two control surfaces the trimming solution is not unique, and the additional constraint of *setting the bow planes deflection to zero* at trimming is imposed.

Stern Plane Deflection at Trimming

The **stern plane deflection at trimming** can be written as

$$\delta_{s0} = \bar{K}_{\delta_s}(\gamma_0, v_{t0}, \zeta_b, \zeta_s)$$

where $\bar{K}_{\delta_s}(.)$ is a nonlinear function of the trimming variables γ_0 and v_{t0} and the control surface areas ζ_b and ζ_s.

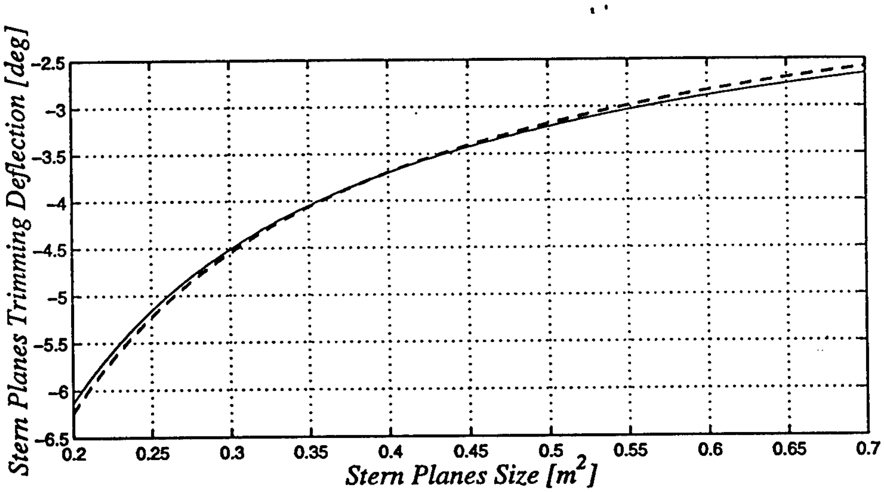


Evolution of $\bar{K}_{\delta_s}()$ for a trimming point characterized by $v_{t0} = 1.5$ m/s and $\gamma_0 = -15$ deg

The stern plane deflection at trimming can be approximated by a first order Taylor expanson in the variable $\frac{1}{\zeta_s}$:

$$\delta_{s0} \simeq \bar{\mathcal{K}}^0_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) + \frac{1}{\zeta_s}\bar{\mathcal{K}}^1_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0})$$

where ζ_{b0} and ζ_{s0} are nominal values about which the expansion is done.



Actual and approximate values of the stern plane deflection at trimming for the case where $v_{t_0} = 1.5$ m/s and $\gamma_0 = 15$ deg. **Solid line**: approximation computed abou the nominal values $\zeta_{b_0} = \zeta_{s0} = 0.4\,\mathrm{m}^2$. **Dashed lines**: actual function obtained for $\zeta_{b_0} \in \{0.3, ., 0.6,\ 0.7\}\mathrm{m}^2$.

Using the approximation

$$\delta_{s0} \simeq \bar{\mathcal{K}}^0_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) + \frac{1}{\zeta_s}\bar{\mathcal{K}}^1_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0})$$

a **constraint on the stern surface area for a given trimming point** $|\delta_s| < \delta_{s_{\max}}$ is locally written as

$$R^+_{\text{trim}}(\zeta_s) := \zeta_s(\bar{\mathcal{K}}^0_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) - \delta_{s_{\max}}) + \bar{\mathcal{K}}^1_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) < 0$$

$$R^-_{\text{trim}}(\zeta_s) := \zeta_s(\bar{\mathcal{K}}^0_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) + \delta_{s_{\max}}) + \bar{\mathcal{K}}^1_{\delta_s}(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0}) > 0$$

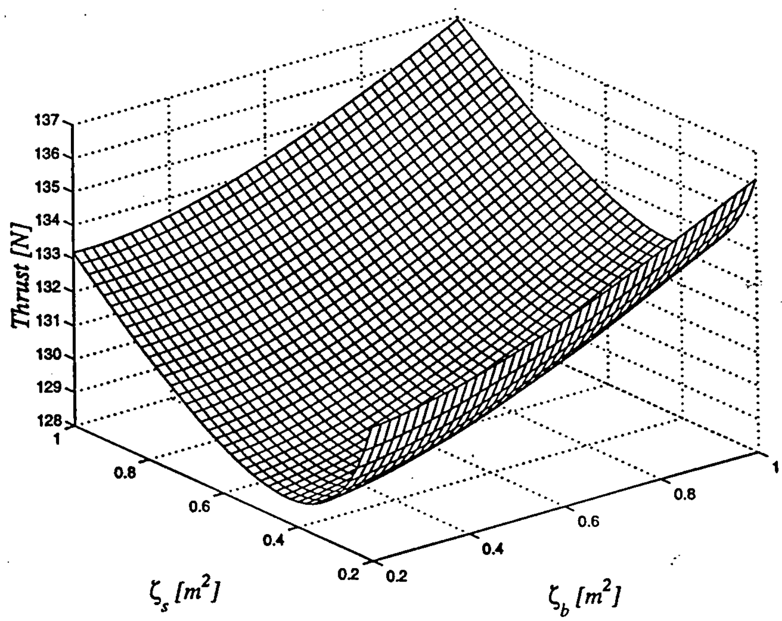
These are **linear inequalities in the stern control surface area** ζ_s.

For small angles of attack, the thrust power P_t at trimming equals $P_t = T_t v_{t_0}$ where

$$T_t = T_t(\gamma_0, v_{t_0}, \zeta_b, \zeta_s).$$

is the thrust required to trim the vehicle.

The figure shows the the function T_t obtained for a trimming trajectory characterized by $v_{t_0} = 1.5$ m/s and $\gamma_0 = -15$ deg



Thrust function shape: an interplay among competing effects:

i) when the **size of the stern control surfaces decreases** they must deflect considerably to achieve trimming a low speed and high flight path angles. This will **increase the total drag and therefore T.**

ii) when the size of the stern control surfaces increases the deflection angles that are required for vehicle trimming decrease; however, since the profile drag is proportional to the surface area, the total drag will eventually increase.

iii) when the size of the bow control surfaces decreases the lift force they generate decreases and the stern control surfaces must deflect to compensate. This will increase the total drag.

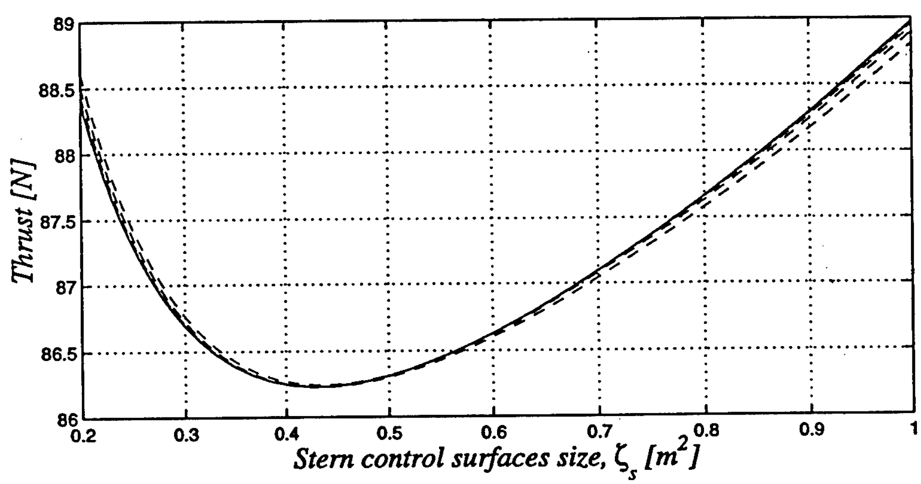
iv) when the size of the bow control surfaces increases the profile drag increases and the total drag will eventually increase.

The thrust T_t at trimming can be approximated as

$$\mathcal{T}_T(\gamma_0, v_{t_0}, \zeta_b, \zeta_s) := \mathcal{T}_{t0}(\gamma_0, v_{t_0}, \zeta_{b_0}, \zeta_{s0}) + \mathcal{T}_{tb0}(\gamma_0, v_{t_0}, \zeta_{b_0}, \zeta_{s0})\zeta_b + \mathcal{T}_{tb1}(\gamma_0, v_{t_0}, \zeta_{b_0}, \zeta_{s0})\zeta_b^{-1} + \mathcal{T}_{ts0}(\gamma_0, v_{t_0}, \zeta_{b_0}, \zeta_{s0})\zeta_s + \mathcal{T}_{ts1}(\gamma_0, v_{t_0}, \zeta_{b_0}, \zeta_{s0})\zeta_s^{-1}$$

by expanding T_t in series of powers of ζ_b, ζ_s, ζ_b^{-1}, and ζ_s^{-1}.

Figure compares the actual and approximate values of T_t for $v_{t_0} = 1.5$ m/s and $\gamma_0 = 15$ deg.

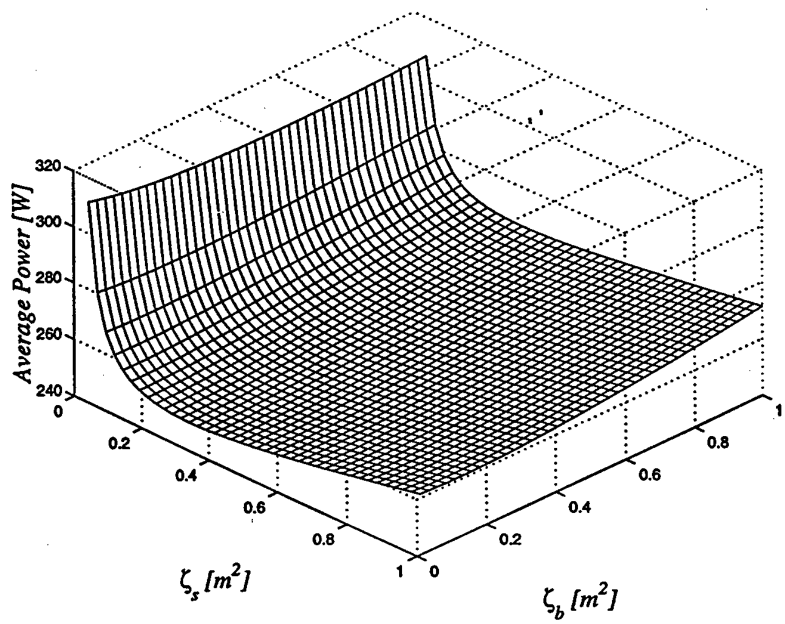


The solid line represents the actual function for $\zeta_b = 0.4\,\mathrm{m}^2$, dashed lines represent the approximation for $\zeta_{s_0} \in \{0.3,\ 0.4,\ 0.5,\ 0.6,\ 0.7\}\mathrm{m}^2$.

Average propulsion power for a given mission:

$$J(\zeta_b, \zeta_s) := \sum_i p_i P_t^i(\gamma_0^i, v_{t0}^i, \zeta_b, \zeta_s),$$

where $P_t^i(.)$ is the power required to trim the vehicle at the flight condition specified by γ_0^i and v_{t0}^i and p_i is the percentage of total mission time that is spent at mission phase i.

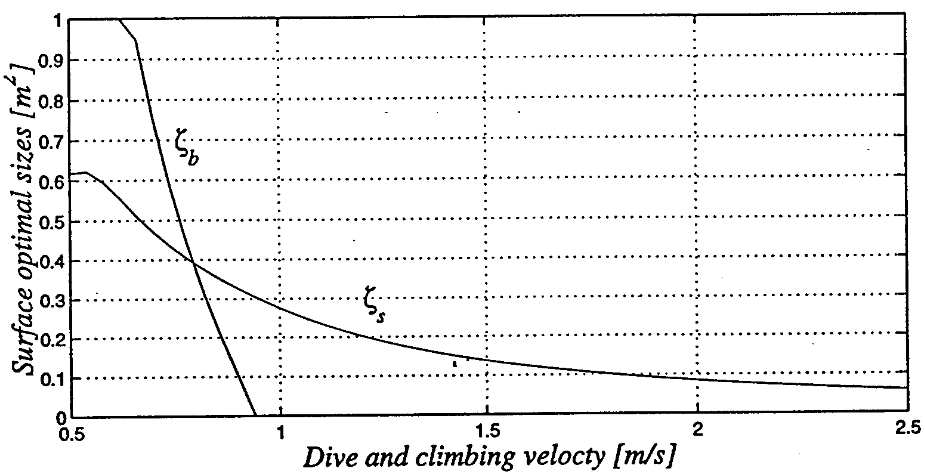


The figure shows the average propulsion power versus the control surface sizes obtained for the mission defined by

$$J(\zeta_b, \zeta_s) = 0.3P_t(-15, 1, \zeta_b, \zeta_s) + 0.4P_t(0, 2.5, \zeta_b, \zeta_s) + 0.3P_t(15, 1, \zeta_b, \zeta_s)$$

Average Propulsion Power versus Surface Size

The figure shows the evolution of the **optimal bow and stern surface sizes** for depth changing maneuvers executed with $|\gamma_0| = 15^o$ and $v_{t_0} \in [0.5,\ 2.5]$ m/s. During level flight, $v_{t_0} = 2.5$ m/s.

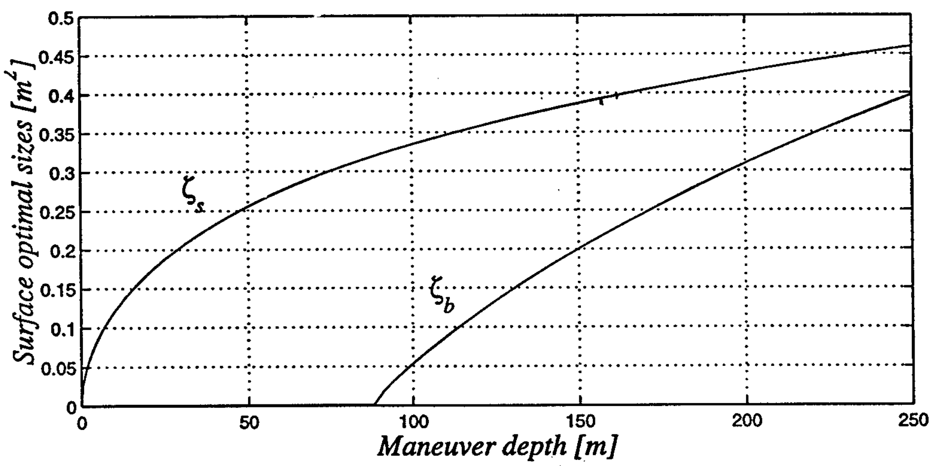


At *lower velocities* the presence of bow planes allows the vehicle to be trimmed at smaller angles of attack, the reduction in drag far outweighing the drag that is induced by augmenting their size.

At *higher velocities*, small control surfaces with small deflections are able to produce the forces and torques required to steer the vehicle along the depth changing maneuvers.

Average Propulsion Power versus Surface Size

Figure was obtained for a three phase mission, where the parameter depth took values in the interval [0, 250] m. The **optimal bow and stern surface sizes** were computed for depth changing maneuvers executed with The figure shows the evolution of the **optimal bow and stern surface sizes** for depth changing maneuvers executed with $|\gamma_0| = 15^o$ and $v_{t_0} \in [0.5,\ 2.5]$ m/s. During level flight, $v_{t_0} =$ 2.5 m/s.



Small depths: trimming achieved with all control surfaces near to zero. This solution minimizes the propulsion power required for the second phase of the mission. As *depth increases*: optimal surface sizes increase to compensate for the drag generated by the control surface deflections necessary to steer the vehicle along the depth changing maneuvers.

The AUV vertical plane model can be written as

$$\dot{\mathbf{x}}_v = F_v(\mathbf{x}_v, \mathbf{u}_v, \zeta_b, \zeta_s)$$

where F_v is a nonlinear function, $\mathbf{x}_v = (\alpha, q, \theta, z)'$ is the state vector, and $\mathbf{u}_v = (\delta_b, \delta_s)'$ is the input vector.

The model is linearized about a trimming point to obtain:

$$\dot{\mathbf{x}}_v = A(\gamma_0, v_{t0}, \zeta_b, \zeta_s)\mathbf{x}_v + B(\gamma_0, v_{t0}, \zeta_b, \zeta_s)\mathbf{u}_v$$

where $A(.)$ and $B(.)$ are matrices that depend on the trimming point (γ_0 and v_{t0}) and on the control surface sizes.

The AUV linear model can be re-written as:

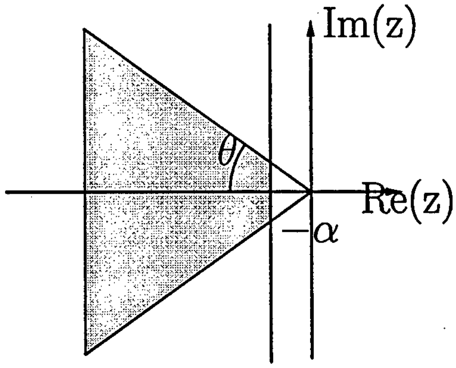
$$\dot{\mathbf{x}}_v = \left[A_0 + \zeta_{\delta_b}A_1 + \zeta_{\delta_s}A_2\right]\mathbf{x}_v + \left[B_0 + \zeta_{\delta_b}B_1 + \zeta_{\delta_s}B_2\right]\mathbf{u}_v$$

where $A_i = A_i(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0})$; $B_i = B_i(\gamma_0, v_{t0}, \zeta_{b0}, \zeta_{s0})$.

Important Fact: the approximate linearizations show a *linear dependence with the variables ζ_b and ζ_s.*

Open Loop Regional Pole Placement Constraints

Open loop generalized stability region



Define

$$A_{ol} = [A(\gamma_0, v_{t_0}, \zeta_b, \zeta_s)_{i,j}], \qquad \text{with} \qquad i,j = 1,2,3$$

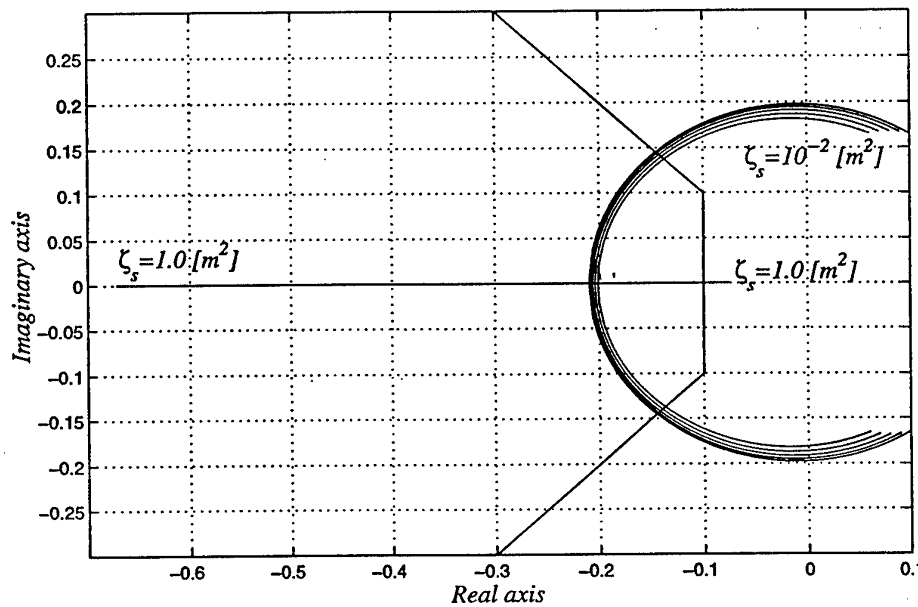
This constraint is satisfied if and only if there exists a symmetric positive definite matrix $X_{ol} > 0$ that verifies the generalized stability Lyapunov inequality

$$R_{ol}(A_{ol}, X_{ol}, \alpha, \theta) < 0$$

where $R_{ol}(.)$ is given by

$$R_{ol}(.) = \begin{bmatrix} \sin(\theta)(A_{ol}X_{ol} + X_{ol}A_{ol}^T) & \cos(\theta)(X_{ol}A_{ol}^T - A_{ol}X_{ol}) & 0 \\ \cos(\theta)(A_{ol}X_{ol} - X_{ol}A_{ol}^T) & \sin(\theta)(A_{ol}X_{ol} + X_{ol}A_{ol}^T) & 0 \\ 0 & 0 & A_{ol}X_{ol} + X_{ol}A_{ol}^T + 2\alpha X_{ol} \end{bmatrix}$$

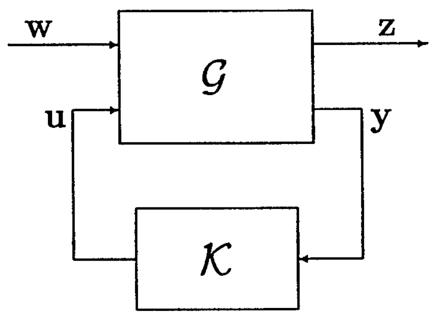
Evolution of the two dominant open loop eigenvalues. Each curve was obtained for a value of $\zeta_b \in \{0.01,\ 0.2,\ 0.4,\ 0.6,\ 0.8,\ 1.0\}$ m^2 and ζ_s within the interval $[0.01\,,1.0]$ m^2, for $\gamma_0 = 0$ and $v_{t_0} = 2.5$ m/s.



Conclusions: small impact of the bow control surface size on overall system open loop stability. This gives the designer an **extra degree of freedom that can be used to improve the vehicle maneuverability**.

Closed loop requirements: the H_∞ set-up

Feedback interconnection.



Let $\mathcal{G}$ admit the realization

$$\begin{aligned}\dot{x} &= Ax + B_w w + B_u u \\ z &= Cx + Du \\ y &= x\end{aligned}$$

Then $\|\mathcal{T}_{\mathbf{zw}}\|_\infty < \gamma$ if and only if there exist a symmetric positive definite matrix $X \in \Re^{n\times n}$ and a matrix $W \in \Re^{q\times n}$ such that the linear matrix inequality (LMI) $R_\infty(X, W, \gamma) < 0$ holds, where $R_\infty(X, W, \gamma)$ is defined by

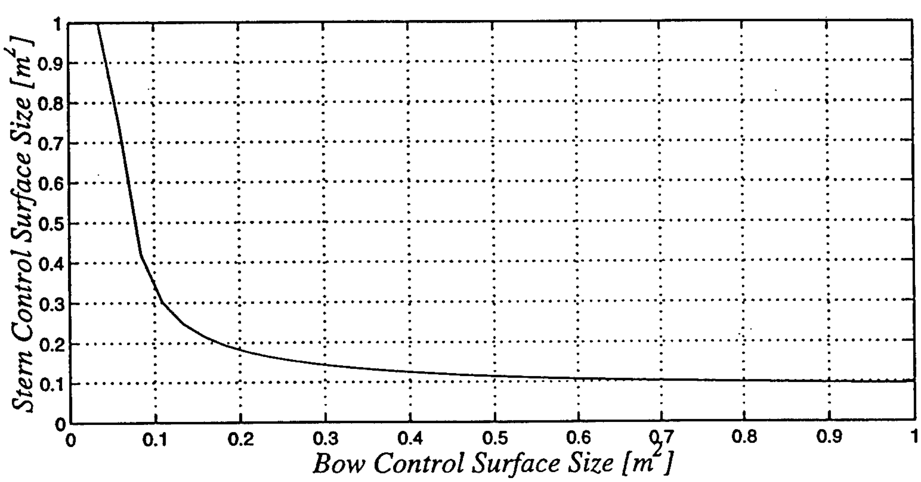
$$\begin{bmatrix} AX_\infty + B_u W + X_\infty A^T + W^T B_u^T & B_w & X_\infty C^T + W^T D^T \\ B_w^T & -\gamma I & D^T \\ CX_\infty + DW & D & -\gamma I \end{bmatrix}. \qquad (1)$$

In case of feasibility a state feedback gain is obtained as $K = WY^{-1}$.

Closed Dynamic Requirements

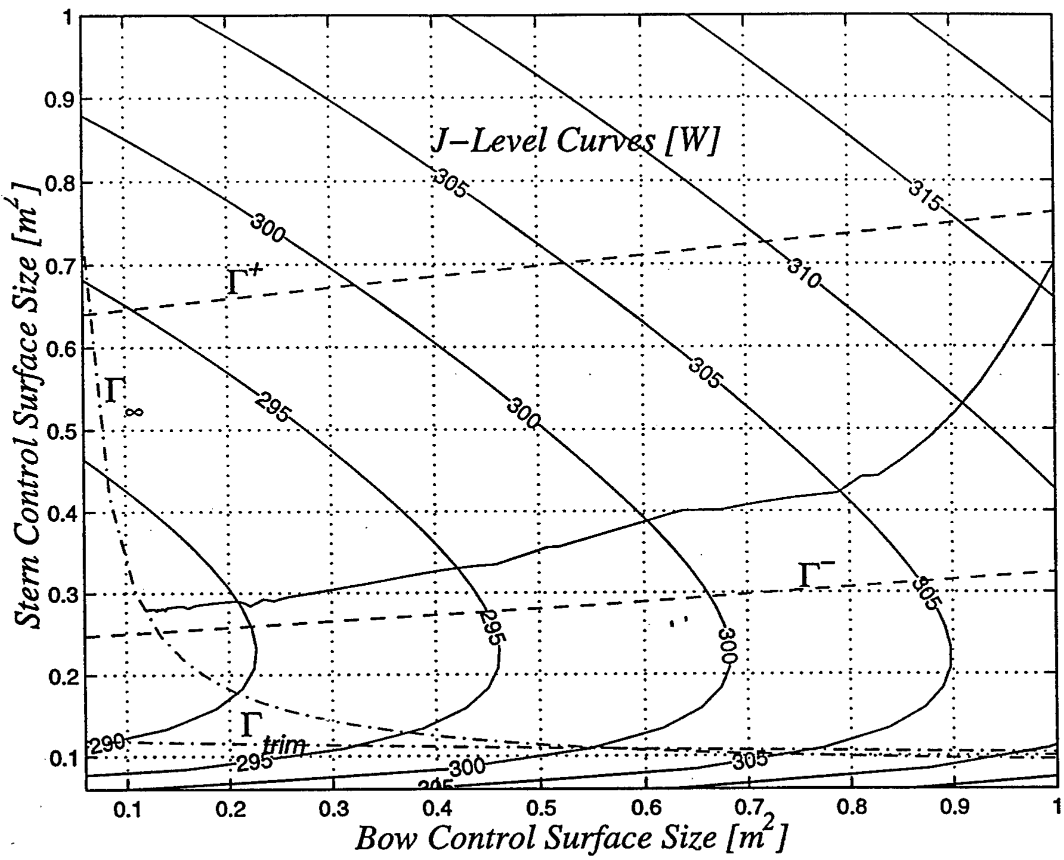
- there should exist a single state feedback controller that simultaneously stabilizes the AUV about all trimming conditions;
- zero steady state in response to depth commands;
- minimum depth command bandwidth of 0.5 rad/s;
- the maximum bow and stern plane control bandwidths should not exceed 2 rad/s,

Figure represents the H_∞ constraint $\gamma \leq 0.8$ boundary of allowable surface sizes



THIS
PAGE
IS
MISSING
IN
ORIGINAL
DOCUMENT

23, 24



In the Figure:

- Γ_∞, Closed loop requirements.
- Γ^+, upper limit of open loop degree of stability requirement.
- Γ^-, lower limit of open loop degree of stability requirement.
- Γ_{trim}, maximum surface deflection at trimming.

Main Results

A new methodology was introduced for the **integrated design of plant parameters and feedback controllers** to meet AUV mission performance requirements with minimum energy expenditure.

Techniques used: Firmly rooted on Linear Matrix Inequalities (LMI) theory. Numerical solutions avaiable with the Matlab LMI Toolbox.

Future work: Extending the technique to address dynamic requirements in the presence of wave disturbances (operation at very small depths).